Image Registration and Optimization in the Virtual Slaughterhouse

Vester-Christensen, Martin

Publication date:
2009

Document Version
Publisher's PDF, also known as Version of record

Image Registration and Optimization in the Virtual Slaughterhouse

Martin Vester-Christensen

Kongens Lyngby 2008
IMM-PHD-2008-206

Technical University of Denmark
Department of Informatics and Mathematical Modeling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

IMM-PHD: ISSN 0909-3192

Preface

This thesis was prepared at the Image Analysis and Computer Graphics group at DTU Informatics and submitted to the Technical University of Denmark - DTU, in partial fulfilment of the requirements for the degree of Doctor of Philosophy, Ph.D., in Applied Mathematics. The project was funded by DTU, Danish Meat Association and the Danish Technical Research Council.

The work herein represents selected parts of the research work carried out in the Ph.D. period. The thesis consists of an introductory, a theoretical part and three research papers.

The work was carried out in a close collaboration with the Danish Meat Research Institute of the Danish Meat Association at the institute in Roskilde, Denmark. Part of the research was conducted at Institut für Mathematik, Universität zu Lübeck, Lübeck, Germany. The project was supervised by Professor Rasmus Larsen - DTU Informatics and Ph.D. Lars Bager Christensen - Danish Meat Research Institute.

Kgs. Lyngby, October 2008

Martin Vester-Christensen

Acknowledgements

This thesis is the product of a collective effort of many people. Each of playing an important role in its completion.

First and foremost my deepest gratitude and respect goes to my supervisor Professor Rasmus Larsen. For accepting me as a Ph.D.-student "back in the day" but also for the whole process of maturing me as a student and person.

The great people at the Danish Meat Research Institute in Roskilde, Denmark. A more open-minded and innovative place is hard to find. I owe an immense debt of gratitude to all the good people there. The willingness to immediately concentrate 100% on my problems has been incredible. The enormous interest in the project has been overwhelming and at times the principal reason for its continuation. Lars Bager Christensen for being a great mentor, innovator and friend. His immense contribution and support is greatly appreciated. Eli V. Olsen for continuously promoting my work and her unconditional support. For introducing me to the many aspects of the pig slaughtering business and showing me the bigger picture. Marchen Hviid for her many contributions, her drive and continued belief in the project. I thank Claus Borggaard, Jesper Blom-Hansen and Niels-Christian Kjærsgaard and many others for great collaboration during the project.

My great friends and colleagues Søren Gylling Hemmingsen Erbou and Mads Fogtmann Hansen. We have truly been comrades in arms. The three years would not have been the same without them.

This Ph.d. project was not half the fun without my great colleagues amongst the staff and Ph.d.-students at the Image Analysis and Computer Graphics group IMM,DTU. Honorable mentions are Sune Darkner for being an exceptional col-

leagues and friend and my office mate Michael Sass Hansen for patiently putting up with the ramblings of a grumpy "old" man. A special thanks goes to Eina and Tove for making the practical issues easy.

The SAFIR group headed by Professor Bernd Fischer at Institut für Mathematik, Universität zu Lübeck for allowing me stay with them for two very funny months in May and June 2007. A truly heartfelt thank you goes to Dr. Jan Modersitzki for accepting me with kindness and friendliness both in Lübeck and in Canada. For sharing of his enormous knowledge and for the patience during my tutoring. That was truly above and beyond the call of duty.

Finally, I extend my love and gratitude to my wife Malene for her complete support and understanding throughout the period. Especially for making my stay in Lübeck possible by being a single-mom for two long months. My daughters Mille and Mathilde for always reminding me of the really important things in life.

Abstract

This thesis presents the development and application of algorithms for the analysis of pig carcasses. Focus is on the simulation and quality estimation of meat products produced in a Danish slaughterhouse. Computed Tomography scans of pig carcasses provide the data used in the application. Image analysis is applied in order to imitate some of the cutting processes found in a slaughterhouse but also to give a quantitative measure of the composition of each carcass.

The basis of the algorithms is non-linear image registration. This method finds the anatomical correspondence between a reference carcass and a template carcass. By iteratively comparing the transformed template with the reference a resulting dense deformation field is found. Propagating a set of landmarks from the reference coordinate system onto the template enables the simulation of slaughtering processes.

Non-invasively estimating the quality of the slaughtering products provides a very valuable tool for use in the slaughterhouse in the future.

Resumé

Denne afhandling beskriver udviklingen og anvendelsen af algoritmer til analyse af svineslagtekroppe. Der fokuseres specielt på simulering og kvalitetsbestemmelse af kød produkter produceret på et dansk slagteri. Computer Tomografi skanning af slagtekroppe udgør data brugt i applikationen. Billedanalyse er anvendt for at efterligne udskæringsprocessen, men også til at give et kvantitativt mål for sammensætningen af de enkelte slagtekroppe.

Algoritmerne er baseret på ikke-lineær billedregistrering. Denne metode finder anatomisk korrespondance mellem en reference slagtekrop og en test slagtekrop. Ved iterativt at sammenligne den transformerede test slagtekrop med referencen findes et resulterende deformationsfelt. Ved at transmittere en række referencepunkter fra reference slagtekroppen til test slagtekroppe muliggøres simuleringen af slagteprocessen.

Ikke-invasiv estimering af kvaliteten af slagteprodukter udgør et meget værdifuldt værktøj til brug i slagterierne i fremtiden.

List of Published Papers

Listed here are peer-reviewed scientific papers and abstracts prepared during the course of the Ph.D. program. The papers included in this thesis is marked with bold typeface of the title.

Journal papers:

- M. Vester-Christensen, S. G. H. Erbou, M. F. Hansen, E. V. Olsen, L. B. Christensen, M. Hviid, B. K. Ersbøll, R. Larsen **Virtual Dissection of Pig Carcasses**. Meat Science, 2008. Submitted.
- S. G. H. Erbou, M. Vester-Christensen, R. Larsen, L. B. Christensen, B. K. Ersbøll. Sparse 3D shape modelling of bone structures *Machine Vision and Applications*, 2008. Submitted.

Conference papers:

- M. Hviid, M. Vester-Christensen. Density of lean meat tissue in pork - measured by CT. *54th International Congress of Meat Science and Technology - ICoMST*, 2008.
- L. B. Christensen, M. Vester-Christensen, C. Borggaard, E. V. Olsen. **Robustness of weight and meat content in pigs determined by CT**. *54th International Congress of Meat Science and Technology - ICoMST*, 2008.
- S. Darkner, M. Vester-Christensen, R.R. Paulsen, R.Larsen. Non-rigid surface registration of 2D manifolds in 3D Euclidian space. *International Symposium on Medical Imaging - SPIE*, 2008.

- M. F. Hansen, S. G. H. Erbou, M. Vester-Christensen, R. Larsen, L. B. Christensen. Surface-to-surface registration using level sets. *Scandinavian Conference on Image Analysis - SCIA*, 2007.
- M. Vester-Christensen , S. G. H. Erbou, S. Darkner, R. Larsen. **Accelerated 3D image registration**. *International Symposium on Medical Imaging - SPIE*, 2007.
- S. Darkner, M. Vester-Christensen, R. Larsen. Evaluating a method for automated rigid registration. *International Symposium on Medical Imaging - SPIE*, 2007.
- S. Darkner, M. Vester-Christensen, R. Larsen, C. Nielsen, R. R. Paulsen. Automated 3D Rigid Registration of Open 2D Manifolds. *From Statistical Atlases to Personalized Models - MICCAI 2006 Workshop*, 2006.
- A. Lyckegaard, R. Larsen, L. B. Christensen, M. Vester-Christensen, E. V. Olsen. Contextual Analysis of CT Scanned Pig Carcasses. *52nd International Congress of Meat Science and Technology - ICoMST*, 2006.

Abstracts:

- M. Vester-Christensen, R. Larsen. Atlas Construction using Fast Image Registration. *Image Analysis in Vivo Pharmachology - IAVP*, 2007.
- S. G. H. Erbou, M. Vester-Christensen, R. Larsen, E. V. Olsen, B. K. Ersbøll. Quantifying Biological Variation. *European Congress of Chemical Engineering - ECCE6 - Special Symposium - Innovations in Food Technology*, 2007.

Contents

Part I

The Virtual Slaughterhouse

Chapter 1

Introduction

The topic of this thesis is, as the title indicates, about slaughterhouses and the slaughtering of pigs. More specifically simulating the process on a computer thus creating a "Virtual Slaughterhouse". A slaughterhouse today is very industrialized and automated. Thousands of pigs are processed each day. The individual carcasses can be cut into a variety of different products. Choosing the right mixture of products for a given carcass is the key to success. However, this decision depends not only on the composition of the carcass but also on the incoming orders from the customers of the slaughterhouse. Being living creatures the pigs are very different anatomically and thus the input into a slaughterhouse is very inhomogeneous. Knowledge of the individual pig is crucial in order to maximize profits and at the same time maintain the level of quality demanded by the customers. The Virtual Slaughterhouse is a tool for providing this knowledge and the means are Computed Tomography scans of the pig carcasses and subsequence image processing.

Computed Tomography(CT) provides an in-vivo view of each carcass, yielding a very detailed level of information about virtually all anatomical regions of the carcass. Complete knowledge of the anatomy provides the means to the potential optimal decision of the use of the individual carcasses. However, the CT data needs to be related to the actual processing of the carcass in the slaughterhouse. Thus, there is need for the Virtual Slaughterhouse.

Slaughtering a pig virtually means cutting a carcass into many products without

it even seeing a knife. Coupling each of these virtual products with a measure of its quality provides the means of choosing the optimal way to process the carcass. The task is to create algorithms capable of mimicking the set of cuts and apply them on a population of carcasses. This is the topic of this thesis.

Image registration is the foundation of the algorithms created in this thesis. It is a well-known discipline used widely in medical imaging and as a consequence many of the algorithms and methods used in the Virtual Slaughterhouse are borrowed from the medical imaging community and literature. However in this thesis the purpose of the application of image registration is remarkably different. A series of virtual cuts are defined in a so-called references carcass. Image registration enables the propagation of these cuts onto a population of pigs thus enabling the subsequent quality measure.

1.1 Thesis Objective

As in any real life application of mathematics and computer science knowledge of the problem domain is required. This work is no exception and it is reflected in the thesis.

The purpose of this thesis is two-fold. It is a theoretical overview of the mathematical disciplines involved in performing image registration. The aim is to provide a more or less self contained guide on the practical implementation of parametric image registration. On the other hand it gives is an overview of some of the concepts and practices involved in the slaughtering of pigs.

Ultimately it is the goal to describe the Virtual Slaughterhouse for the automatic evaluation of the quality of cuts of meat.

1.2 Thesis Overview

The thesis is divided into four parts.

Part **I** is the introductory part. It provides the background needed to better understand the problem domain in which the thesis is placed. Chapter 2 describes the Danish meat industry with emphasis on the pig slaughtering industry and it describes Virtual Slaughterhouse. A review of the kind of data and the data acquisition is described in chapter 3.

Part **II** is a theoretical description of the image registration algorithms used.

It provides the theoretical background and practical implementation issues involved in parametric image registration. Chapter 4 introduces image registration and provides an overview of the literature. In chapter 5 the concept of images and image transformation is explained. Chapter 6 contains the bulk of the part and describes image registration as a large scale optimization problem. It revolves around the Gauss-Newton algorithm and contains the solution to practical implementation issues such as dealing with the very large matrices.

Part **III** describes the actual application of the algorithms on carcass data. Chapter 7 is a self-contained description of the algorithms implementing the automatic quality grading using virtual cuts. It is a demonstration of the power in applying the use of CT scans and image analysis on a practical problem such as estimating the quality of certain cuts of meat.

Part **IV** is a collection of published contributions written during the project. They cover different aspects of the concepts found in the thesis. Chapter 8 describes an experiment demonstration the robustness of using the CT as a quantitative measuring tool. The results show that CT offers a robust measurement technology with a very small sensitivity to the different settings of the scanning protocol. Chapter 9 describes the creation of a linear model of the relation between voxels and the full weight of the half carcass. This knowledge enables the determination of the lean meat percentage. The lean meat percentage is a measure of the quality of the meat in a carcass and used to predict the quality of individual cuts. Chapter 10 describes the creation of an alternative parametric image registration algorithm than that described in part two.

Finally appendix A contains a view of two of the demonstration programs produced during the project.

Chapter 2

CT in the Meat Industry

The topics of this thesis are connected to the meat industry. More specifically the pig slaughtering industry. The Danish slaughterhouses are undergoing a change with the use of advanced technology in the slaughtering process. In recent years the use of Computed Tomography(CT) as modeling and measuring tool in connection with the slaughtering of pigs has been investigated. This project is denoted The Virtual Slaughterhouse. The name covers the building of computerized mathematical models of many aspects of the processes going on in a slaughterhouse. Using these models a pig carcass can be virtually slaughtered on a computer before it even sees a knife and the results can be evaluated. Thus providing some of the building blocks of a decision support system to be used in the slaughterhouses.

2.1 The Danish Meat Industry

The Danish meat industry is one of the largest and most important industries in the country, with the pig industry being by far the largest part. The annual export of pig meat is roughly 30 billion DKK, around 4.0 billion EUR, and represents about 5 pct. of the total annual export [114]. As such the story of the pig industry is for the most part a story of success. However in recent years, especially 2007 a year with increasing price on feed, the profits have decreased.

Figure 2.1: Examples of automation in a Danish slaughterhouse. Carcasses hanging on bars in the equalization room(top left). Top right and lower left images depicts before and after a split into three parts. The lower right image depicts a slaughter robot splitting the middle part into loin and belly. Images are courtesy of Danish Meat Association.

Competition in the industry is hard. The entrance of competitors from low wage countries on export markets with a strong Danish presence further increases the pressure. In 2005 a Danish slaughterhouse worker had an average hourly wage of 218 DKK, while the average hourly wage of a German slaughterhouse worker was 98 DKK [65]. This mismatch, and other added expenses due to a strict Danish legislation on health and safety issues, makes the production of pig meat very expensive in a Danish slaughterhouse. One solution already implemented in the slaughterhouses is an increased automation of some of the labor intensive processes. Figure 2.1 shows some of the automated tasks. As a means to ensure the slaughterhouses continued survival in Denmark there is an increased interest on the maximization of the profit of each individual carcass [75]. This thesis is part of this effort and focusses on the optimal choice of end products for which the carcass is used.

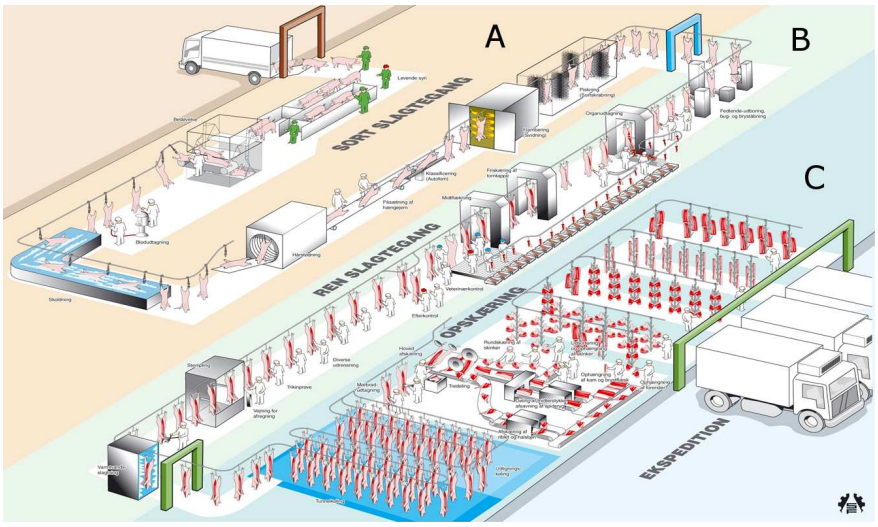


Figure 2.2: Graphical rendering of the product flow in a slaughterhouse. The pigs enter in section A which is the killing zone, where they are stunned, killed by sticking and debled. They continue onto the slaughterline in section B where the carcass is split in halves and intestines, heart etc. are removed. Section C is the cutting line in which the carcasses are cut into products.

2.1.1 From Pig to Product

To set the stage for the rest of the thesis this section contains a description of the journey a carcass undertakes from the kill to the end product. It is an adaptation of a description of the overall product flow in Danish slaughterhouses found in [75] but is included here for completeness.

Figure 2.2 depicts the product flow in a slaughterhouse. Three sections comprise the slaughterhouse. Section A is the killing zone. Section B contains the slaughtering lines, and section C the cutting lines. The process commences in section A, where the pigs are stunned, killed by sticking and debled. To remove the hairs the carcasses are scalded in a hot water bath, de-haired, and singed. The singing has an added effect on the tenderness of the rind. The carcasses then continue into section B which is the slaughtering line.

On the slaughtering line the carcasses are cut open and the intestines, heart, liver, kidney etc. are removed and inspected for disease and abnormalities. A slaughtering robot splits the carcass in halves by cutting along the spine, but leaves the two halves still connected by the jawbone. The carcasses are then cleaned and weighed. The fat layer thicknesses in the fore end, middle and ham

part of the carcass are measured and the lean meat percentage is estimated for the total carcass. These parameters enters the classification algorithms for the subsequent sorting of the carcass. At a modern slaughterhouse there is four slaughtering lines with a rate of 350 pigs per hour constituting a total of $1,400$ pigs per hour in total.

After slaughtering the carcasses enter an equalization room with the purpose of lowering the carcass temperature over a period of at least 16 hours. It is also here that the carcasses are sorted. Due to logistical reasons a carcass is placed on a bar containing similar pigs. A bar consist of roughly 80 carcasses. The upper left image in figure 2.1 shows carcases hanging from bars. The actual bar a pig is assigned to is based on a forecast of how the pig is to be used in the subsequent cutting. The sorting is primarily based on the valuable middle part and the ham. The task for the planners in the slaughterhouse is to decide the optimal use of the carcasses placed on a bar considering their classification but also the incoming orders.

After cooling the carcasses enter section C in which initially the head and tenderloin are cut of. Each half of the carcass is split into three parts. The front end, the middle and the leg(ham) part. Being the most valuable the middle and ham can undergo further sorting based on weight. All parts are then cut based on their sorting group and at the end of the cutting line the products are placed on stands containing 20 items. The stands are placed in a holding area awaiting further cutting into specific products which are packed and shipped to customers.

In this process knowledge of the individual pigs is important. Today the slaughterhouses obtain this information in the classification system. The estimation of the lean meat percentage and subsequent classification is based on sparse samples of the depth of the fat layer in the carcass. This measure estimates with high probability the composition of the entire carcass, however complete knowledge is not attained. Basing a classification system on CT-scans of each carcass would give almost complete information of the composition of the pig. This could yield a classification with an even higher probability. But even more importantly it could provide the data for a wider range of measurements than just the depth of the fat layer. One example is direct feedback to the farmer of the state of his pigs. A more differentiated payment scheme could even be devised awarding pigs with certain qualities. A computerized simulation of the different cuts on each carcass as described later in this thesis is another example. The result could be a more elaborate sorting scheme fully optimizing the use of each carcass. As it is today the utilization of this knowledge is somewhat hindered by the logistics surrounding the equalization room. However increasing the knowledge of the pigs can further the existing sorting process and thus optimize the carcass use.

2.2 Pig Carcass Classification - The Lean Meat Percentage

Classification of pigs has been in use, in some form other, in the Danish slaughterhouses since the 1930s and is used to objectively determine the composition of the carcass. Since the 1970s the lean meat percentage has been the classifying parameter. Lean meat is defined as the percentage of the total carcass weight constituted by the lean meat. The Danish method was adopted by the EU and since 1989 it has been mandatory to classify with an EU approved method [96]. The Danish Pig Classification Authority inspect the slaughterhouses each week to ensure that classification is done consistently in all slaughterhouses. This makes sure that the settlement to the farmer is consistent regardless of the choice of slaughterhouse.

Classification and classification equipment is governed by the EU through the Management Committee for Pig Meat [28]. The lean meat percentage can be estimated using various types measuring equipment and all classifying equipment has to be approved by the committee. Common for all measuring equipment is the calibration towards the lean meat content determined by a dissection. Dissection is a process of discerning all meat in a carcass from the rest. Thus a manual lean meat percentage can be obtained. In the EU a common reference method was developed in 1995 [134]. However it is shown that there are regional differences when using this method [92, 97].

Consequently investigations in the use of 3D imaging technology as a method to obtain the lean mean content of a carcass has been undertaken in recent years. Most notably is the EUPIGCLASS project [38] in which the use of CT and MRI was investigated.

This leads to the proposal of using CT as a reference method. Here a classification into fat, meat or bone of the individual voxels in the carcass volume is done using a pixel classification method. Models for predicting the weight of the various tissue types using CT is developed and thus the lean meat percentage can be determined. The method and an experiment is described in [131] and in chapter 9 of this thesis. This enables the prediction of weight and lean meat percentage in subsections of a scanned carcass as well and an application utilizing this is described in [130] and in chapter 7.

2.3 Previous Work and Applications

Several studies with regards to the use of CT and MRI to determine the composition of meat, fat and bone in carcasses has been performed. Some of the earliest studies include Skjervold *et al.* [112] and Allen and Vangen [3]. A survey of the estimation of the composition of pig carcasses using digital imaging techniques including CT is found in Szabo *et al.* [119]. In [88, 89] Monziols *et al.* predicts the lean meat content in MRI images of carcasses. Furthermore investigation regarding the partial volume effect is also carried out. Another study is performed in the EUPIGCLASS project [38].

In [132] Vestergaard *et al.* uses mathematical morphology on CT scans on salt dried hams to estimate the salt content.

Related to this field is research using image analysis on 3D images of sheep and lambs. In [68, 76] Kongsro investigates the use of CT and spectral methods to determine the lean meat content of lamb carcasses. Stanford *et al.* [115] reviews the use of CT and MRI amongst others to determine the composition of lamb carcasses. In [91] Navajas *et al.* uses segmentation to calculate muscle volume.

2.4 The Virtual Slaughterhouse

At its heart The Virtual Slaughterhouse is a collection of algorithms and software which is developed during the last $3-4$ years. Several applications have been researched ranging from the statistical modeling of femoral bones, through virtual dissection to virtual cuts. Listed in this section are some of the research topics investigated in the Virtual Slaughterhouse.

In [130](chapter 7) Vester-Christensen *et al.* investigates the use of image registration as a tool for measuring the quality of virtual cuts. Image registration provides the means to propagate a set of cuts defined in an atlas onto a population of carcasses and estimate the yield of the different cuts. In [83] Lyckegaard uses image registration to infer statistics of a specific of cut of the middle part of the carcass. In [56] Hansen segments individual muscles in the middle part using level sets. Furthermore investigation of the quality of a specific cut using heuristics is also done.

In [35–37] Erbou *et al.* researches statistical modeling of the shape of femoral bones. This provides a tool for the builders of slaughter robots. The statistical model provides knowledge of the distribution of the dimensions of the femoral bone which is useful in the determination of tolerances needed when building a robot. In [57] Hansen *et al.* performs surface registration of the femoral bones

as a means to build the shape model.

In [50] Grønlund utilizes some of the software produced to manually segment muscles and model the lean meat percentage.

In [71–75] Kjærsgaard investigates the economic effects of the optimization of the carcasses under the current logistical limitations in the slaughterhouses. He finds that a profit increase is obtainable and worth pursuing but the amount is hindered by the current logistical situation concerning the equalization room.

Chapter 3

Imaging

The image data produced in The Virtual Slaughterhouse stems from a wide variety of experiments conducted. Ranging from live pigs over half pig carcasses to chickens and cows. In addition to the imaging data usually other types of data are also recorded. A very large experiment described below and in chapter 9 involved manually dissecting 50 half carcasses into fat, meat and bone tissue. In [66] is described an experiment in which fat, water, protein and collagen content is measured in samples of meat. Furthermore a subjective human grading is often linked to the scanned volume, e.g. a visual grading of the quality of certain products, enabling the development of models to automate this.

3.1 Half Carcasses

The main data set used in this thesis consist of scans of 299 half pig carcasses. The experiment is a part of a large calibration trial conducted in the spring of 2008. See chapter 9 for a more thorough description of the trial and experiment. The pigs in the set are chosen to be representative of the Danish pig population when ready for slaughter with respect to age, weight and fatness. The weight ranges from approximately 60 kg to about 100 kg and they are slaughtered at an approximate age of six months.

A half pig carcass is the product of the pig carcass being split along the spine. In most Danish slaughterhouses this is an automated process in which a robot uses a saw to perform the split. Figure 3.1 depicts a back splitting robot. After the splitting the head and toes are cut off and the carcass is cooled. The day after slaughtering the left side of the carcass is ready for scanning. A half carcass ready for scan is shown in figure 3.2.

Finally the carcass placed on the skin side and scanned in the CT-scanner, see figure 3.3.

3.2 Computed Tomography 101

The imaging modality used to produce the data is Computed Tomography. The technology originates from Sir Godfrey Hounsfield in 1972 and is widely used both as a medical examination tool but also in the industry as a inspection tool [67, 70]. A CT-scanner produces a series of two dimensional cross sections of an object. It functions by transmitting X-rays from a source while rotating it around the object and recording the transmitted radiation with detectors placed in a surrounding gantry. Every slice consists of a grid of voxels. The voxels have been crossed by numerous X-rays during the scan and the attenuation of the corresponding tissue can be reconstructed from the detector readings. Figure 3.4 illustrates this principle. Each voxel is assigned a relative value of the average attenuation of the enclosed tissue compared to the attenuation of water. The scanner is calibrated prior to scanning using a so-called phantom setting the scale such the value 0 corresponds to the attenuation of water and -1000 to that of air. Performing this calibration enables comparison of voxel values across subjects which is very important when using the scanner as a quantitative measuring tool.

A histogram over the voxel values of a half pig carcass is seen in figure 3.5. A more in-depth discussion of Hounsfield values for tissues found in pig carcasses is found in chapter 9.

3.3 Image Data

The carcasses are scanned using a refurbished GE HiSpeed CT/i single-slice scanner capable of isotropic voxels. An average prepared half carcass is around 130 cm in length which would yield a $512 \times 512 \times 1300$ volume using $[1 \times 1 \times 1]$ mm voxels. However, time, monetary concerns and computer processing power constrains the volumes size. The voxel dimensions are lowered to $1 \times 1 \times 10$

Figure 3.1: A back-splitting robot seen from the front. Courtesy of Danish Meat Association.

Figure 3.2: A half carcass prepared and ready for scanning. Courtesy of Danish Meat Association.

Figure 3.3: A carcass wrapped in low density plastic about to be scanned. Courtesy of Danish Meat Association.

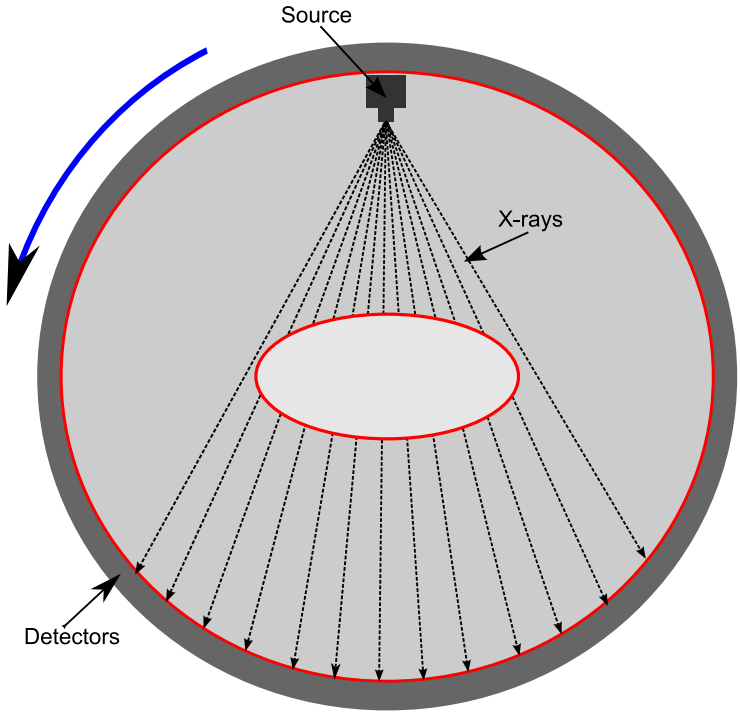


Figure 3.4: The principle of Computed Tomography.

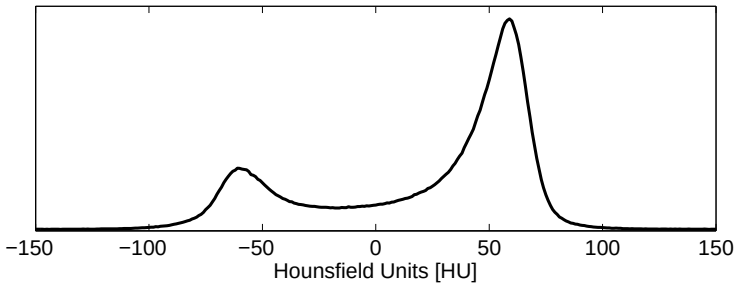


Figure 3.5: Histogram of voxels values in a typical pig, clamped in the range $[-150; 150]$. The left peak corresponds to fat, the right peak to meat. Bone is found as the tail in the right side of the histogram.

Figure 3.6: An axial slice of the carcass volume located in the middle part.

mm resulting in a reduction in the number of slices needed to cover the whole carcass. This, however, results in a reduction in the resolution in the sagittal and coronal planes. Figure 3.6 shows a slice obtained in the middle part of the carcass, while figure 3.7 depicts a montage of every second slice in the carcass volume.

Other useful visualizations of the volume data borrows from medical imaging. Being a 3D volume two other views of a location in the carcass is needed to obtain complete visual coverage. Figure 3.8 depicts the midsagittal and midcoronal planes of a carcass. Notice the jagged edges due to the reduced resolution. Other visualizations tries to capture the 3D nature of the data. Figure 3.9 shows a simple method visualizing orthogonal slice planes. The volume effect is obtained by arranging the axial, sagittal and coronal views in a three dimensional space.

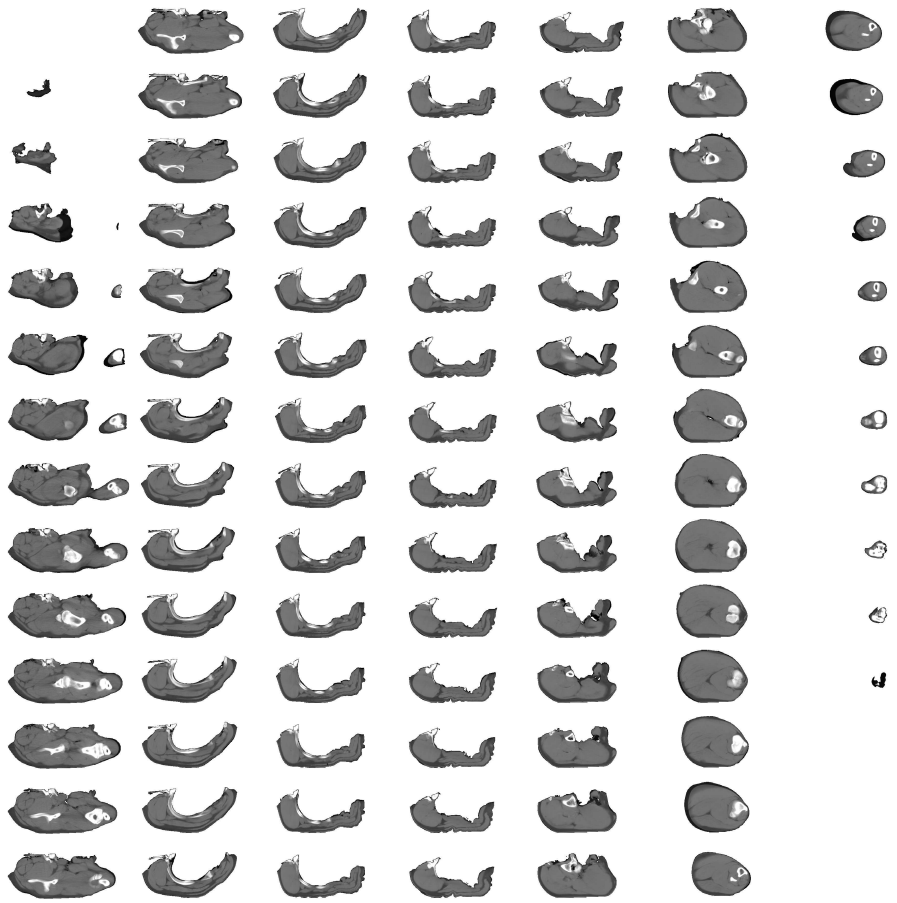


Figure 3.7: A montage view of every second slice in a CT volume of a carcass.

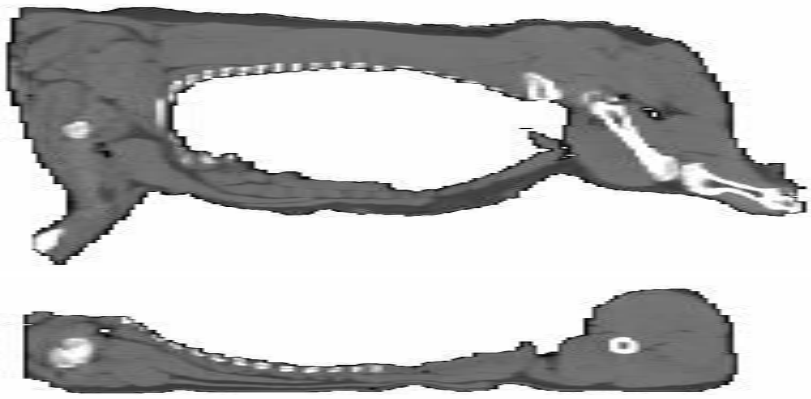


Figure 3.8: Sagittal and coronal views of a carcass.

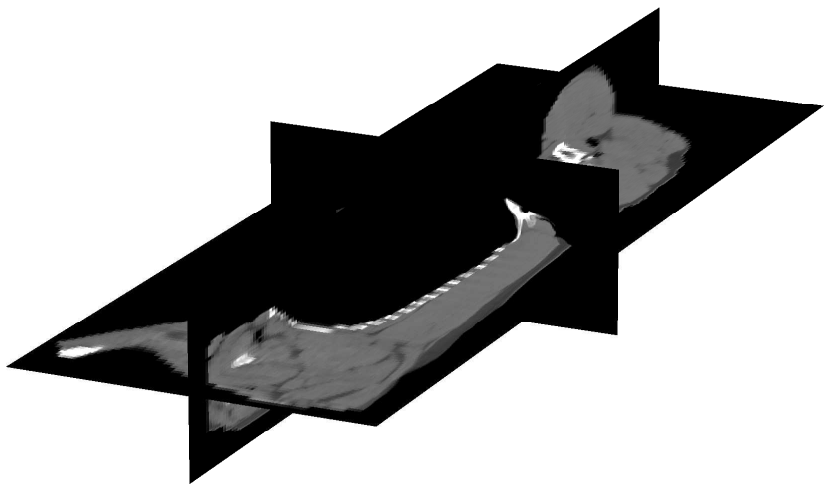


Figure 3.9: Volumetric visualization of a half carcass using orthogonal slice planes.

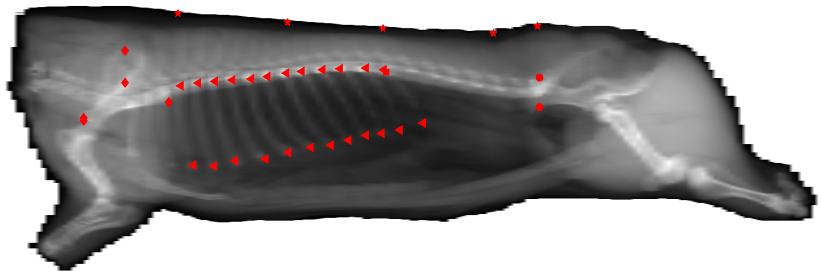


Figure 3.10: X-Ray view of a carcass with landmarks overlain.

3.4 Anatomical Landmarks

To navigate in the CT volume and later to compare the registration accuracy a set of anatomical landmarks is defined. The landmarks are placed in locations that are easily identified and repeatable across the population of pigs in the data set. However to utilize the landmarks in the applications described later, the landmarks are also placed at reference points in the carcass which are used by the butcher or slaughter robot when cutting the carcass into products. Figure 3.10 depicts an X-Ray view of a carcass with the landmarks overlain. A total of 37 landmarks is used. In figure 3.11 a 3D surface rendering of the ribs and scapula is shown with the landmarks overlain.

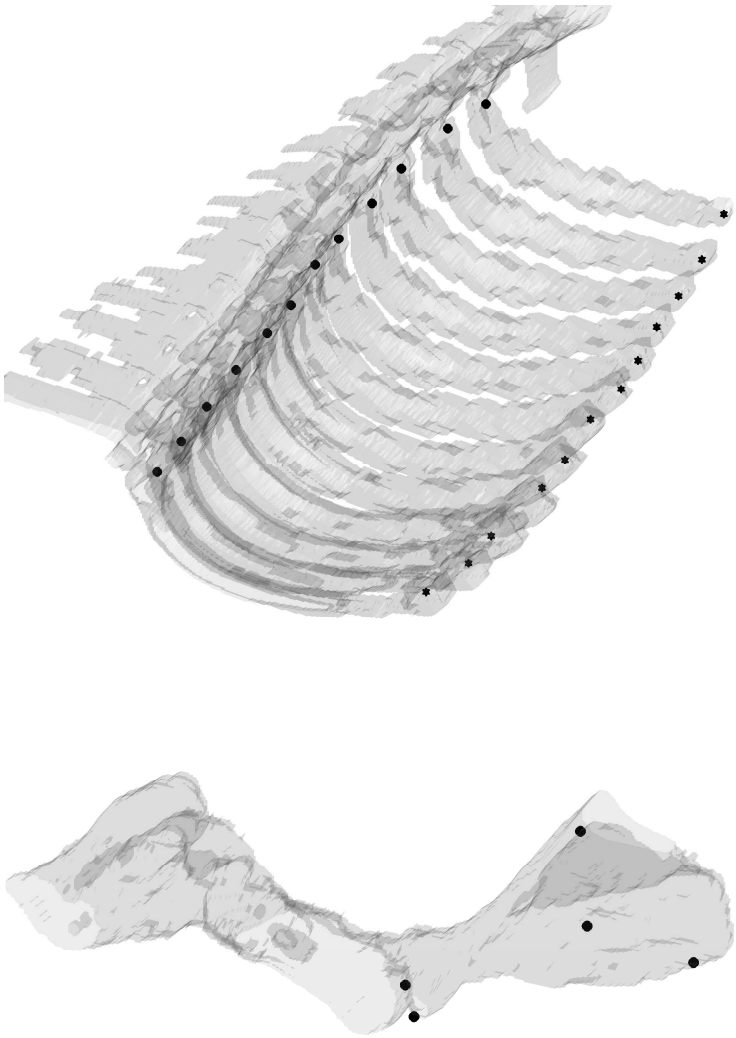


Figure 3.11: Annotation scheme for the twelve most posterior ribs and the scapula.

Part II

Algorithms for Image Registration

Chapter 4

Introduction

The task of image registration can be phrased as the following [53]:

> Given a so-called reference image R and a so-called template image T, find a reasonable geometrical transformation such that a transformed version of the template image becomes similar to the reference image.

This definition also lists the disciplines involved in image registration and incidently they are the topics of the chapters in this part of the thesis,

- Image transformation. Change to the geometry of the image.
- Similarity. Measure the goodness of the registration.
- Optimization. Control the transformations to maximize similarity.
- Regularization. Allow only reasonable transformations.

Each of the items listed is a research area in its own right, and as such there exist a number of different algorithms and methods for each discipline.

In the course of this Ph.d. project a framework for 2D and 3D parametric image registration has been created. It is manifested in an implementation in Matlab

capable of registration of large volumes. The following chapters are not by any means a comprehensive tutorial on image registration, but rather an overview of the algorithms and concepts utilized in this framework.

4.1 Algorithms and Applications

4.1.1 Registration Algorithms

Image registration is a very active research area. Especially in the field of medical imaging but also in a variety of other applications. The applications and subsequent problems to solve in this thesis are very similar to problems found in medical imaging. Luckily this leaves a host of prior research in medical imaging to lean upon. Consequently most literature found in this overview stems from the medical imaging community.

Several methods and applications of image registration have been published. Books covering the topic of image registration are amongst others; Gothasby [49], Modersitzki [87] and Hajnal *et al.* [55]. Thorough surveys of the many algorithms and applications can be found by Zitova and Flusser [137], by Pluim *et al.* [100], by Maintz and Viergever [86], by Lester and Arridge [81] and by Brown [18]. Basically the task is to find the optimal dense field transforming a template image into a reference image. Several approaches exist.

A group of algorithms solves the problem using a variational approach. Two parts are involved. One part models of the deformation of the underlying tissue regularizing the deformations. The other part is a measure quantifying the similarity of the images. These parts forms the basis of a system of non-linear partial differential equations(PDE). In 1981 Broit [17] based his solution on the linear theory of elasticity modeling the tissue as an elastic material and since other works [6, 15, 22, 40, 45, 87] has followed a similar approach.

Spearheaded by Christensen [22] was an approach where the image was regarded as a viscous fluid. The advantage over the elastic approach is that the fluid has no "memory" and as such can model larger deformations. Amongst others, algorithms have been made by Bro-Nielsen and Gramkow [16] and by Henn and Witsch [62].

Another approach using diffusion as a regularizer is proposed in Fischer and Modersitzki [41]. An adaptation of this work is found in [42].

Solving the regularized problem without forming the PDE's is found in the works of Haber and Modersitzki [53, 54]. A similar method is found in the

algorithm by Hellier *et al* [60] however the similarity measure and regularizer is formulated using robust statistics. The Demons algorithm by Thirion [122] [123] is modeling the deformations via a diffusion process based on optical flow and Gaussian smoothing of the deformation field. Recently the Demons algorithm has been adapted and modernized in the work of Vercauteren *et al.* [126].

In the above mentioned algorithms no parameters are involved in the description of the deformation field. However a host of algorithms parameterizing this field exist.

A very popular algorithm is based on Free Form Deformations and is described in Rueckert *et al* [108, 111] and is based on the parametrization of the deformation field using B-splines. A similar method also using B-splines is found in the thesis by Kybic [77]. In the works by Ashburner [4, 5] basis functions based on the discrete cosine transform is used. Cootes *et al.* [32] ensures that the transformation is diffeomorphic by constraining basis functions based on a cosine kernel.

Several software packages are available for image registration. Among them are the popular AIR package [135], the tools NA-MIC [99], and the SPM package [5].

4.1.2 Applications

Image registration is used in a wide variety of applications. Most similar to the applications found in this thesis are those based on atlases.

In the works of Cootes *et al.* [30, 32] and Twining *et al.* [127] image registration is used to simultaneously build an atlas and automatically find the optimal landmark annotation. A groupwise cost function based on Minimum Description Length [33] is used for simultaneous registration.

In a similar approach [117] Studholme uses a groupwise mutual information cost function to simultaneously register a population of brains for spatial normalization.

In [12] Blezek et al. uses a technique named atlas stratification. Here they try to infer from the image data whether one or multiple atlases is needed to best describe the population.

In [94] Olafsdottir uses image registration to study craniofacial anomalies. Statistical deformation models are built and statistics on the craniofacial abnormalities of Crouzon Mice are inferred.

Atlases have many applications but the most similar to the application in this

thesis is atlas based segmentation [10, 102, 104, 105]. Here the anatomical regions in the atlas is labeled. This labeling is then propagated onto the templates segmenting one or several structures.

Chapter 5

Images and Image Transformation

5.1 Images

By nature of the acquisition method digital images are discrete, but with the help of an interpolation scheme they can be viewed as being continuous. Interpolation enables the evaluation of the image function anywhere in the domain. A wide variety of interpolation schemes exist [121].

Solving the problem of image registration is however done using discrete entities. Thus a discretization of the continuous image is needed. This is done by sampling the image on a grid of points. For a $D-$dimensional image of size $n_1 \times \ldots \times n_D$ voxels, the ith grid point is defined as,

$$\boldsymbol{x}_i = (x_i^1, \ldots, x_i^D)^\top. \tag{5.1}$$

A grid is thus a collection of grid points and is here defined as a vector. A 3D grid of size $n_1 \times n_2 \times n_3$ is a vector of size $3N \times 1$ with $N = n_1 n_2 n_3$. In this work a grid has a certain lexicographical order. Grouping the ordinates x_i^1, x_i^2

and x_i^3 of all the grid points into three auxiliary vectors,

$$\begin{aligned} \boldsymbol{x}^1 &= \left(x_1^1 \dots x_N^1\right)^\top \\ \boldsymbol{x}^2 &= \left(x_1^2 \dots x_N^2\right)^\top \\ \boldsymbol{x}^3 &= \left(x_1^3 \dots x_N^3\right)^\top, \end{aligned} \tag{5.2}$$

the 3D grid $\boldsymbol{X}$ is defined as,

$$\boldsymbol{X} = \left[\boldsymbol{x}^{1\top}\ \boldsymbol{x}^{2\top}\ \boldsymbol{x}^{3\top}\right]^\top. \tag{5.3}$$

Sampling an image on a discrete grid is also a task for the interpolation scheme. A sample from a continuous image $\mathcal{T}$ at the grid point $\boldsymbol{x}_i$ is written as,

$$T\left(\boldsymbol{x}_i\right) = \mathcal{T}\left(\boldsymbol{x}_i\right). \tag{5.4}$$

A discrete image is denoted,

$$T(\boldsymbol{X}) = \mathcal{T}(\boldsymbol{X}) \tag{5.5}$$

where $\boldsymbol{X}$ indicates samples on an entire grid.

In the literature the dominating interpolation method is linear interpolation. The force of linear interpolation is its ease and resulting speed of computation requiring only eight samples in 3D. However it is only C^0 continuous and is as such not differentiable. Another widely used scheme is based on cubic B-splines and is C^2 continuous, but significantly more computationally demanding. See [120] for a discussion of the merits of B-spline interpolation in image registration.

When working with images it is important to use physical coordinates. Fortunately data obtained from medical imaging devices, such as a CT-scanner contain information about the size of voxels, the orientation of the patient and so on. The relation between a 3D physical coordinate $\boldsymbol{x}_i$ and a voxel coordinate $\hat{\boldsymbol{x}}_i$ is simply,

$$\boldsymbol{x}_i = \begin{bmatrix} h_1 & 0 & 0 \\ 0 & h_2 & 0 \\ 0 & 0 & h_3 \end{bmatrix} \begin{bmatrix} \hat{x_i^1} \\ \hat{x_i^2} \\ \hat{x_i^3} \end{bmatrix}, \tag{5.6}$$

where the origo and orientation of the physical image is coincident with the voxel image. h_1,h_2 and h_3 are the physical dimensions of a voxel. Thus the grid coordinates are given in metric units.

Figure 5.1 depicts a slice of a CT volume. The top image is the result of the continuous image being sampled with a fine grid and thus with a small voxel size. In the bottom image the grid is much coarser.

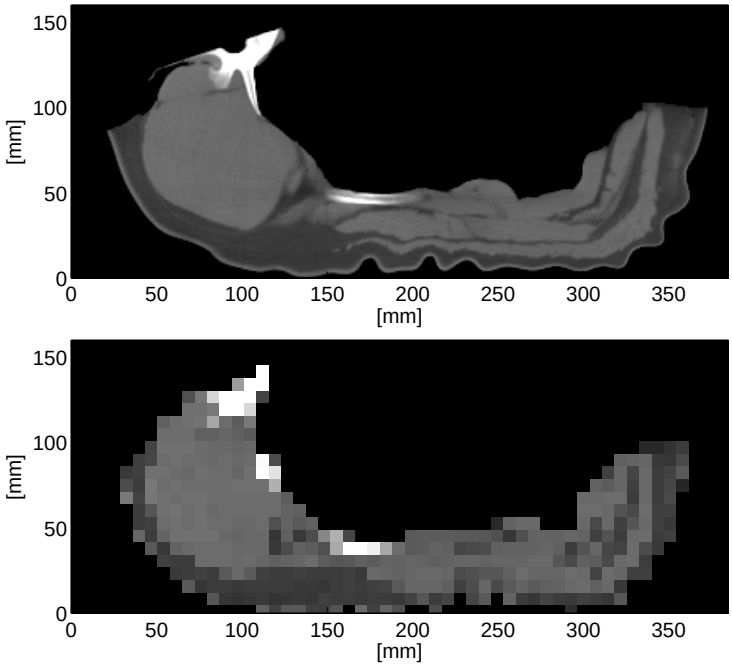


Figure 5.1: Images with different voxel dimensions.

5.2 Image Transformations

In order to make two different images similar one or both has to change. This is the task of image transformation. It involves transforming the underlying grid points by some function,

$$\boldsymbol{y}_i = \phi\left(\boldsymbol{x}_i\right), \tag{5.7}$$

and the transformed image is then obtained by evaluating the continuous image $\mathcal{T}$ with a transformed grid $\boldsymbol{Y}$,

$$T(\boldsymbol{Y}) = \mathcal{T}(\boldsymbol{Y}), \tag{5.8}$$

using an interpolation scheme.

The transformation can also be seen as a sum of an identity part and a deformation part,

$$\begin{aligned} y_i^1 &= x_i^1 + u^1(\boldsymbol{x}_i) \\ y_i^2 &= x_i^2 + u^2(\boldsymbol{x}_i) \\ y_i^3 &= x_i^3 + u^3(\boldsymbol{x}_i), \end{aligned} \tag{5.9}$$

where $u^d(\boldsymbol{x}_i)$ is the displacement of a grid point $\boldsymbol{x}_i$ in the dth dimension. Thus for each grid point there is a corresponding displacement vector,

$$\boldsymbol{u}_i = \left(u^1(\boldsymbol{x}_i)\ u^2(\boldsymbol{x}_i)\ u^3(\boldsymbol{x}_i)\right)^\top. \tag{5.10}$$

All the displacement vectors of the entire grid form a field. In the literature this field is known as the deformation field or the displacement field. Using the same lexicographical ordering rules as for a grid $\boldsymbol{X}$ indicated in equation 5.2 and 7.1 the displacement vector can be stacked in a vector $\boldsymbol{U}$. Thus the transformed grid can also be written as,

$$\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{U}. \tag{5.11}$$

Figure 5.2 shows an example of an image transformation. The bottom left image shows the image $T(\boldsymbol{X})$ before transformation. The top left plot show the grid $\boldsymbol{X}$ with the displacement vectors overlain. The bottom right figure depicts the transformed image $T(\boldsymbol{Y})$ produced by interpolating the left image using the resulting grid $\boldsymbol{Y}$ shown in the top right figure.

It is possible to parameterize the transformation in equation 5.9 using linear

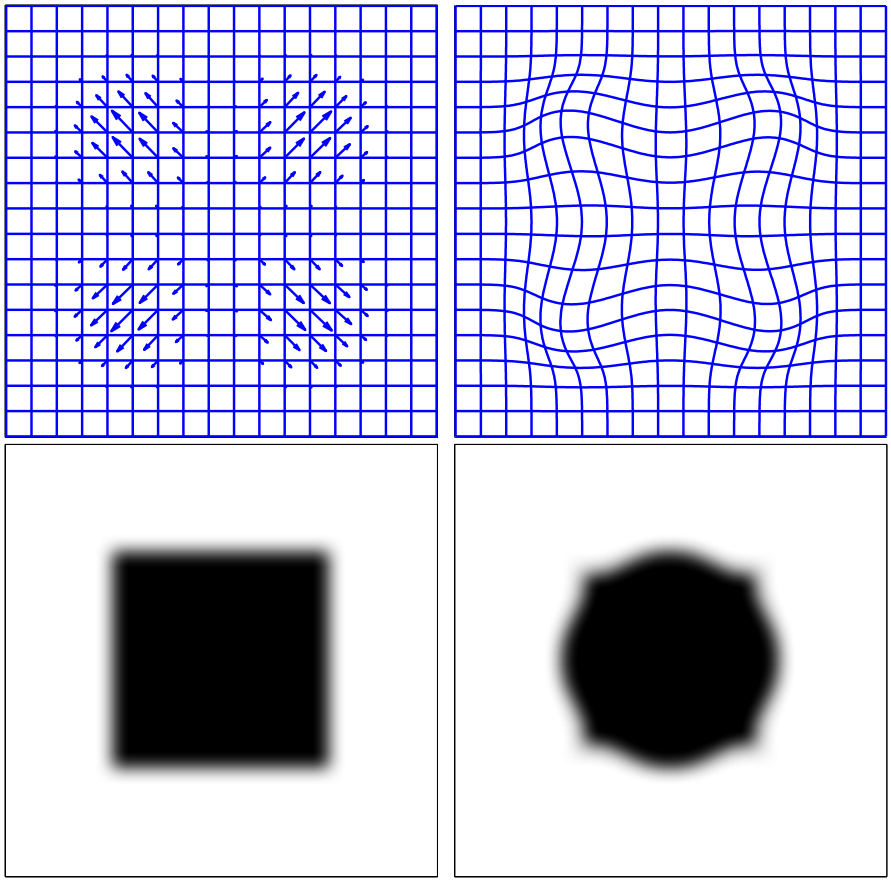


Figure 5.2: Example of an image transformation turning a square into a circle. The top left figure depicts the grid $\boldsymbol{X}$ overlaid with the displacement vectors. Top left is the resulting grid $\boldsymbol{Y}$. In the bottom row is shown the image before $T(\boldsymbol{X})$ (left) and after $T(\boldsymbol{Y})$ (right) the transformation.

combinations of basis functions. The expression then becomes,

$$\begin{aligned} y_i^1 &= x_i^1 + \sum_j^m b_j(\boldsymbol{x}_i) w_j^1 \\ y_i^2 &= x_i^2 + \sum_j^m b_j(\boldsymbol{x}_i) w_j^2 \\ y_i^3 &= x_i^3 + \sum_j^m b_j(\boldsymbol{x}_i) w_j^3, \end{aligned} \tag{5.12}$$

where w_j^d is the weight of the jth basis function in the dth dimension. A vast amount of possible basis function exist and are used in the literature [5, 29, 108]. Encoding the basis function evaluations in a matrix $\boldsymbol{Q}$ with the element at the ith row in the jth column being $q_{ij} = b_j(\boldsymbol{x}_i)$ enables simplification of the above expression. Considering the entire grid $\boldsymbol{X}$ the transformation can be written as,

$$\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{I}_3 \otimes \boldsymbol{Q}\boldsymbol{w}, \tag{5.13}$$

where $\boldsymbol{I}_3$ is a unit matrix of rank 3 and $\boldsymbol{w}$ is a vector of the parameters as in,

$$\boldsymbol{w} = \left(\boldsymbol{w}^{1\top}\ \boldsymbol{w}^{2\top}\ \boldsymbol{w}^{3\top}\right)^\top, \tag{5.14}$$

and

$$\begin{aligned} \boldsymbol{w}^1 &= \left(w_1^1 \dots w_m^1\right)^\top \\ \boldsymbol{w}^2 &= \left(w_1^2 \dots w_m^2\right)^\top \\ \boldsymbol{w}^3 &= \left(w_1^3 \dots w_m^3\right)^\top. \end{aligned} \tag{5.15}$$

In general we want the deformation field to be smooth. If the application is simply to make the images as similar as possible the regularity of the resulting grid is of no concern. Figure 5.3 depicts such a situation. A rectangle is transformed completely into a circle. However the resulting grid shows a large amount of fold over. In this case the grid is degenerate and the inverse transformation cannot be recovered. Statistics on the deformations would be useless and registration based segmentation would not be possible. A measure often used in the literature [53, 61] to determine if fold-over has occurred is the determinant of the Jacobian of the displacement vectors.

5.3 Affine Transformations

In many image registration applications it is desirable to remove effects from the image acquisition process. For instance in an experiment calculating statistics

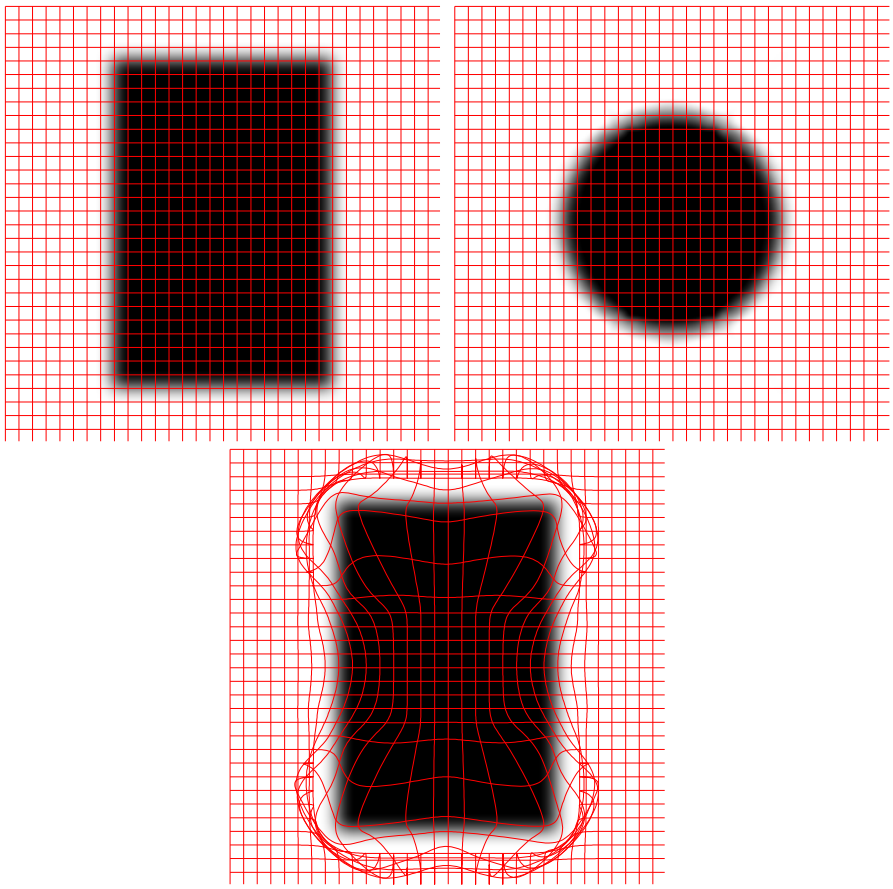


Figure 5.3: A transformation of a rectangle. Top left is the starting rectangle with the grid $\boldsymbol{x}$ overlain. Top right is the resulting circle sampled using the grid $\boldsymbol{y}$. The bottom figure depicts the resulting grid $\boldsymbol{y}$.

on the shape of objects effects such as how the object is placed in the scanner should not be a factor. To remedy this the images are preregistered using linear transformations. The most flexible linear transformation is the affine transformation, which can model rotation, translation, scaling and shearing. For a 3D grid it is written as,

$$\begin{bmatrix} y_i^1 \\ y_i^2 \\ y_i^3 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ a_5 & a_6 & a_7 & a_8 \\ a_9 & a_{10} & a_{11} & a_{12} \end{bmatrix} \begin{bmatrix} x_i^1 \\ x_i^2 \\ x_i^3 \\ 1 \end{bmatrix} \tag{5.16}$$

or expressed using basis functions,

$$\begin{aligned} y_i^1 &= x_i^d + \sum_{j=1}^{4} w_j^1 q_j \left(\boldsymbol{x_i}\right) \\ y_i^2 &= x_i^d + \sum_{j=1}^{4} w_j^2 q_j \left(\boldsymbol{x_i}\right) \\ y_i^3 &= x_i^d + \sum_{j=1}^{4} w_j^3 q_j \left(\boldsymbol{x_i}\right), \end{aligned} \tag{5.17}$$

where w_j^d is the jth coefficient for the corresponding basis function q_j in the dth dimension. The very simple basis functions are,

$$q_1 \left(\boldsymbol{x_i}\right) = x_i^1, \quad q_2 \left(\boldsymbol{x_i}\right) = x_i^2, \quad q_3 \left(\boldsymbol{x_i}\right) = x_i^3, \quad q_4 \left(\boldsymbol{x_i}\right) = 1. \tag{5.18}$$

Considering the entire grid and putting in matrix notation this yields,

$$\boldsymbol{Y} = \boldsymbol{X} + \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{Q} \end{bmatrix} \boldsymbol{w} = \boldsymbol{X} + \boldsymbol{I}_3 \otimes \boldsymbol{Q}\boldsymbol{w}, \tag{5.19}$$

$\boldsymbol{Y}$ is the transformed grid, $\boldsymbol{Q}$ is a $N \times 4$ matrix of basis function evaluations and $\boldsymbol{w}$ is a 12×1 vector of coefficients. Figure 5.4 shows an example off an affine transformation.

5.4 B-Spline Transformations

A type of transformation capable of modeling non-linear deformations is based on B-splines. B-splines are widely used as a basis in image registration [77, 108]. B-splines has very nice properties such as minimal support for a given order of

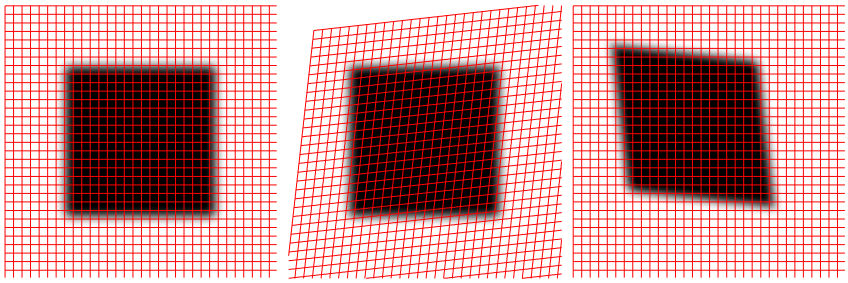


Figure 5.4: Affine transformation of a square. The left figure depicts the square and the grid before transformation, while the middle figure shows the square with the transformed grid. The right figure shows the square sampled using the transformed grid.

approximation, and they are maximally continuous [121, 128]. Thus a cubic B-spline is C^2 continuous and are non-zero only in four neighboring knot intervals.

Using a non-recursive definition [121] the following 1D cubic B-splines basis functions can be derived,

$$b_j(x_i) = \begin{cases} (2+x_i)^3 & \text{for } -2 \leq x_i < -1 \\ -(3x_i+6)x_i^2+4 & \text{for } -1 \leq x_i < 0 \\ (3x_i-6)x_i^2+4 & \text{for } 0 \leq x_i < 1 \\ (2-x_i)^3 & \text{for } 1 \leq x_i < 2 \\ 0 & \text{otherwise} \end{cases} \tag{5.20}$$

The basis for a cubic B-spline with equidistantly placed knots is simply shifted copies of the 'mother' function in equation 5.20. Figure 5.5 depicts the basis functions of a cubic B-spline using free boundary conditions.

B-splines are separable and so it is easy to extend the basis functions to 3D. The transformation is written as

$$\begin{aligned} y_i^1 &= x_i^1 + \sum_{j=1}^{m_1}\sum_{k=1}^{m_2}\sum_{l=1}^{m_3} b_j(x_i^1)b_k(x_i^2)b_l(x_i^3)w_{jkl}^1 \\ y_i^2 &= x_i^2 + \sum_{j=1}^{m_1}\sum_{k=1}^{m_2}\sum_{l=1}^{m_3} b_j(x_i^1)b_k(x_i^2)b_l(x_i^3)w_{jkl}^2 \\ y_i^3 &= x_i^3 + \sum_{j=1}^{m_1}\sum_{k=1}^{m_2}\sum_{l=1}^{m_3} b_j(x_i^1)b_k(x_i^2)b_l(x_i^3)w_{jkl}^3, \end{aligned} \tag{5.21}$$

where m_1,m_2 and m_3 are the number of equidistantly placed knots for the splines. In matrix notation using the tensor product property,

$$\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{I}_3 \otimes \boldsymbol{Q}\boldsymbol{w} = \boldsymbol{X} + \boldsymbol{I}_3 \otimes \boldsymbol{Q}^3 \otimes \boldsymbol{Q}^2 \otimes \boldsymbol{Q}^1\boldsymbol{w}, \tag{5.22}$$

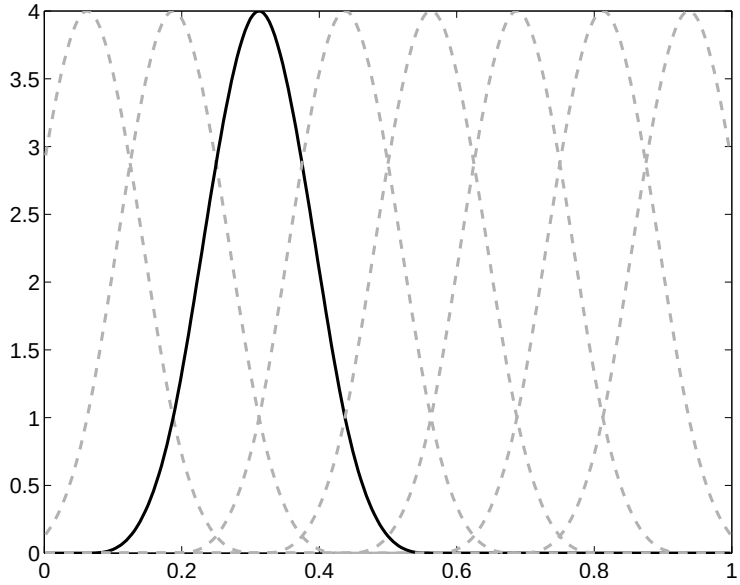


Figure 5.5: The basis functions of a uniform cubic B-spline are just shifted copies of the same 'mother'-function.

where $\boldsymbol{Q}^d$ is a $n_d \times m_d$ matrix with rows consisting of the evaluation of the spline function, $\boldsymbol{w}$ is a $m_1 m_2 m_3 \times 1$ vector of parameters Figure 5.2 depicts a B-spline based transformation.

Chapter 6

Image Registration - a Large Scale Optimization Problem

6.1 Similarity Measurements

Succeeding in making images similar requires a quantitative measure of the goodness of the registration. It also involves an algorithm to control the image transformations such that similarity is maximized. In this thesis an optimization approach is chosen. Using parametric transformations it naturally becomes an optimization of the similarity with regards to the parameters. Using optimization community lingo the similarity measure is often called a cost function or an objective function.

Similarity is usually determined by comparing values of corresponding voxels. A host of different similarity measures exist. One of the most dominant, especially in the medical imaging community, is Mutual Information [44, 85, 133] which is suited for comparison of multi-modality images [118] or when contrast agents are applied. Another often used measure is based on the normalized cross-correlation coefficient [26, 80].

In this thesis the focus is on images acquired using a CT-scanner. The scanner is calibrated with a phantom as part of the preparation. This means that X-ray attenuation of corresponding tissue types are directly comparable across scans.

A similarity measure suited for this situation is the sum-of-squared-differences (SSD) measure. It assumes that the differences in value of corresponding voxels are independently, identically normally distributed. SSD measures the Euclidean distance between grey values of corresponding voxels in a reference image R and a template image T,

$$\begin{aligned} D(\boldsymbol{Y}) &= \frac{h}{2}\sum_i \left(T(\boldsymbol{y}_i) - R(\boldsymbol{x}_i)\right)^2 \\ &= \frac{h}{2}\|T(\boldsymbol{Y}) - R(\boldsymbol{X})\|_{L2}^2, \end{aligned} \tag{6.1}$$

with $h = h_1 h_2 h_3$ being the product of the voxel dimensions. Maximizing image similarity thus requires a minimization of SSD. Since the minimization is driven by parametric image transformations, equation 6.1 is phrased as a function of $\boldsymbol{w}$,

$$D(\boldsymbol{w}) = \frac{h}{2}\|T(\boldsymbol{w}) - R\|^2, \tag{6.2}$$

where the subscript L_2 indicating the Euclidean norm is left out and $R = R(\boldsymbol{X})$. A key benefit of minimizing SSD is that it is a least-squares problem. Solving least-square problems is part of a vast amount of data fitting and modeling applications and several tried and tested algorithms exist [93].

6.1.1 Non-Linear Least-Squares

When minimizing equation 6.2 it would be preferable to find the value of $\boldsymbol{w}$ for which the measure has the lowest value possible. In other words to find a global minimizer. In most image registration problems this is however not feasible. Thus, we have to settle for a local minimizer, defined by Nocedal and Wright [93] as the value $\boldsymbol{w}^*$ for which $D(\boldsymbol{w}^*) \leq D(\boldsymbol{w})$ for $\boldsymbol{w}$ belonging to some neighborhood of $\boldsymbol{w}^*$. In other words $\boldsymbol{w}^*$ is at the bottom of a valley in the cost function landscape. To determine that a point $\boldsymbol{w}^*$ is a minimizer of equation 6.2 we need information of the function in the neighborhood. A way to obtain this information is by Taylor expanding D,

$$D(\boldsymbol{w} + \boldsymbol{s}) = D(\boldsymbol{w}) + \boldsymbol{s}^\top \nabla D(\boldsymbol{w}) + \frac{1}{2}\boldsymbol{s}^\top \nabla^2 D(\boldsymbol{w})\boldsymbol{s} + \mathcal{O}(\|\boldsymbol{s}\|)^3, \tag{6.3}$$

where ∇D is the gradient and $\nabla^2 D$ is the Hessian matrix.

Finding the minimizer is often done by an iterative algorithm. From a starting point the parameters $\boldsymbol{w}$ are updated with a vector $\boldsymbol{s}_k$ pointing in the direction of lower values of the function D,

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \boldsymbol{s}_k, \tag{6.4}$$

and stops when certain criteria are met. To determine the update $\boldsymbol{s}_k$ two dominating schemes exist, namely line search and trust region methods. The algorithms in this thesis use the line search strategy and so it is described below. Trust region methods are by no means inferior but for the sake of compactness they are left out. An extensive introduction can be found in [93].

The benefit of SSD being a least-squares problem comes in the composition of the gradient and Hessian. For ease of presentation equation 6.2 can also be phrased in terms of the vector of residuals $\boldsymbol{r} = T(\boldsymbol{w}) - R$,

$$D(\boldsymbol{w}) = \frac{h}{2}(T(\boldsymbol{w}) - R)^\top (T(\boldsymbol{w}) - R) = \frac{h}{2}\boldsymbol{r}^\top \boldsymbol{r}. \tag{6.5}$$

Thus the gradient is,

$$\nabla D(\boldsymbol{w}) = h\nabla \boldsymbol{r}^\top \boldsymbol{r} = hJ(\boldsymbol{w})^\top \boldsymbol{r}, \tag{6.6}$$

where $J(\boldsymbol{w})$ is the Jacobian which is the matrix of first partial derivatives of the residuals,

$$J(\boldsymbol{w}) = \begin{bmatrix} \frac{\partial r_1}{\partial w_1} & \cdots & \frac{\partial r_1}{\partial w_m} \\ \vdots & \ddots & \\ \frac{\partial r_n}{\partial w_1} & \cdots & \frac{\partial r_n}{\partial w_m} \end{bmatrix}. \tag{6.7}$$

The corresponding Hessian is,

$$\nabla^2 D(\boldsymbol{w}) = h\nabla \boldsymbol{r}^\top \nabla \boldsymbol{r} + h\nabla^2 \boldsymbol{r}\boldsymbol{r} = hJ(\boldsymbol{w})^\top J(\boldsymbol{w}) + h\nabla^2 \boldsymbol{r}\boldsymbol{r}. \tag{6.8}$$

Because the first part is by far the most dominating most algorithms skip the evaluation of the second derivatives. Furthermore, in image registration problems it is generally not recommended to evaluate the second derivatives of images [87]. When the Jacobian is evaluated it can be seen that the first part of the Hessian $J(\boldsymbol{w})^\top J(\boldsymbol{w})$ comes for "free".

6.2 The Gauss-Newton Method

A local minimizer has the lowest local value of $D(\boldsymbol{w})$ and so the line search strategy is simply a matter of going downhill in the cost function landscape. Two problems exists. The first is finding a descent direction $\boldsymbol{s}_k$, and the second is to determine the size α of the step to take.

An obvious path would be to go in the direction of fastest decay of the cost function or more accurately in the direction of steepest descent as in $-\nabla D$.

However as stated in any textbook on optimization this will be extremely slow on all but the simplest problems. In general a descent direction is defined as,

$$\boldsymbol{s}_k^\top \nabla D(\boldsymbol{w}_k) < 0. \tag{6.9}$$

i.e. a direction with an angle in the open interval $]-\pi/2, pi/2[$ to $-\nabla D$. A much better choice would be to include information from the Hessian and so we end up with the Newton direction,

$$\boldsymbol{s}_k = -\nabla^2 D(\boldsymbol{w}_k)^{-1} \nabla D(\boldsymbol{w}_k), \tag{6.10}$$

assuming that the second derivative exists and is positive definite.

Utilizing the benefits of least-squares problems equation 6.10 leads to a very simple iterative algorithm namely the Gauss-Newton method. It revolves around solving the linear system also known as the normal equations [93],

$$\boldsymbol{J}_k^\top \boldsymbol{J}_k \boldsymbol{s}_k = -\boldsymbol{J}_k^\top \left(T\left(\boldsymbol{w}\right) - R\right), \tag{6.11}$$

where $\boldsymbol{J}_k$ is the Jacobian $J(\boldsymbol{w}_k)$.

As seen the Gauss-Newton algorithm only involves calculating the residuals and the Jacobian. For parametric image registration the Jacobian is,

$$J(\boldsymbol{w}) = \frac{\partial \left(T\left(\boldsymbol{Y}\right) - R\right)}{\partial \boldsymbol{w}}, \tag{6.12}$$

which by using the chain-rule is found to be,

$$\boldsymbol{J}_k = \boldsymbol{\nabla T Q} = \left[\boldsymbol{\nabla T}^1 \ \boldsymbol{\nabla T}^2 \ \boldsymbol{\nabla T}^3\right] \boldsymbol{I}_3 \otimes \boldsymbol{Q}_3 \otimes \boldsymbol{Q}_2 \otimes \boldsymbol{Q}_1, \tag{6.13}$$

where each $\boldsymbol{\nabla T}^d$ is a diagonal matrix containing the partial derivatives $\partial T(\boldsymbol{y}_i)/\partial x^d$ of the image T.

The listing in algorithm 6.1 sketches the Gauss-Newton algorithm for image registration. To stop the algorithm certain criteria can be set. These criteria should prevent the algorithm from running when no progress is made. Often used are; the decrease in cost function $D(\boldsymbol{w}_{k-1}) - D(\boldsymbol{w}_k) < \epsilon_D$, the size of the Jacobian $\|J_k\| < \epsilon_J$, the difference in parameters $\|\boldsymbol{s}_{k-1} - \boldsymbol{s}_k\| < \epsilon_s$ or of course the number of iterations $k > k_{max}$.

6.3 Step Length

For a minimizer $\boldsymbol{w}^*$ two conditions exist. A first order condition on the gradient,

$$\nabla D(\boldsymbol{w}^*) = 0, \tag{6.14}$$

Algorithm 6.1 Gauss-Newton algorithm

1: Set starting guess $\boldsymbol{w} = \boldsymbol{w}_0$
2: **while** NOT STOP **do**
3: Transform the grid $\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{Q}\boldsymbol{w}$.
4: Compute $T(\boldsymbol{Y})$ and $\nabla T(\boldsymbol{Y})$.
5: Get $\boldsymbol{s}_k$ by solving the linear system in equation 6.11.
6: Calculate step length α using algorithm 6.2 or 6.3.
7: Update $\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \alpha\Delta\boldsymbol{w}_k$
8: **end while**

requiring the minimizer to be at a stationary point. A second order condition on the Hessian $\nabla^2 D$,

$$\nabla^2 D(\boldsymbol{w}^*) \text{ must be positive semidefinite,} \tag{6.15}$$

requiring the function to be convex at the position of the minimizer.

Choosing the right step length parameter α is an important task and has impact on the convergence rate of the optimization algorithm. It is basically a one dimensional problem, namely minimizing,

$$\phi(\alpha) = D(\boldsymbol{w} + \alpha\boldsymbol{s}_k) \tag{6.16}$$

with respect to α. However solving the problem exactly can be computationally very costly and unnecessary if we relax on the quality of the minimization. Two conditions known as the *Wolfe conditions* exist and it can be shown that steps satisfying these yield acceptable convergence rates [93].

The first Wolfe condition guarantees sufficient decrease in the cost function,

$$\phi(\alpha) \leq \phi(0) + c_1\phi'(0)\alpha \tag{6.17}$$

where $c_1 \in (0, 1)$. This ensures a decrease of the cost function at least a factor c_1 more than the one predicted by the slope of the cost function at the starting point. A simple method is listed in algorithm 6.2 in which α meets the criteria set by equation 6.17. It backtracks from a starting value until a stopping criteria is enforces. Of course the initial value for $\hat{\alpha}$, τ, c_1 and the number of line search iterations can be set by the user. The backtracking algorithm also makes sure that the step taken is not too short. It backtracks from steps that are too long until an acceptable $\hat{\alpha}$ is found. The optimal value of α is within an the interval $(\alpha_l/\tau, \alpha_l)$ where α_l is the last value rejected.

The second Wolfe condition is important in algorithms where steps of lengths larger than 1 are required. The second Wolfe condition is stated below,

$$|\phi'(\alpha)| \leq c_2|\phi'(0)|, \tag{6.18}$$

Algorithm 6.2 Backtracking line search algorithm

1: Set $\hat{\alpha} = 1$, $\tau = \frac{1}{2}$, $c_1 = 10^{-6}$ and $LS_{max} = 10$
2: **for** $l \leq LS_{max}$ **do**
3: **if** $D(\boldsymbol{w}_k + \hat{\alpha}_l \boldsymbol{s}_k) > D(\boldsymbol{w}_k) + c_1 \hat{\alpha}_l \nabla D(\boldsymbol{w})^\top \boldsymbol{s}_k$ **then**
4: $\hat{\alpha}_{l+1} = \tau \hat{\alpha}_l$.
5: **else**
6: $\alpha = \hat{\alpha}_l$ and `stop`
7: **end if**
8: **end for**

where $c_2 \in (c_1, 1)$. This is a condition on the size of the gradient at a prospective location. Only locations where the size of the gradient is a factor smaller than at the start location are accepted. This ensures that the steps have a certain minimal length. Line search algorithms implementing both Wolfe conditions are more elaborate than simple backtracking. Most algorithms are performing a *soft* line search. The basic idea is to find an interval $[\alpha_{lo}, \alpha_{hi}]$ in which a step length satisfying the Wolfe conditions is known to be located. Then reduce that interval until an appropriate step length satisfying both Wolfe conditions is found.

A line search algorithm [43, 93] satisfying both Wolfe conditions is listed in algorithm 6.3. It utilizes the knowledge that an interval $[\alpha_{lo}, \alpha_{hi}]$ is sure to contain a viable step length if either of three conditions is satisfied,

- α_i violates the sufficient decrease conditions.
- $\phi(\alpha_i) \geq \phi(\alpha_{i-1})$.
- $\phi(\alpha_i)' \geq 0$.

If such an interval is found the optimal value of α is determined. It is done by fitting a model function and finding the minimizer of it. Here a quadratic function of the form,

$$\hat{\phi}(\alpha) = a_3(\alpha - \alpha_{lo})^2 + a_2(\alpha - \alpha_{lo}) + a_1, \tag{6.19}$$

is fitted using only information from the interval endpoints. Since $\hat{\phi}(\alpha_{lo}) = \phi(\alpha_{lo})$, $\hat{\phi}'(\alpha_{lo}) = \phi'(\alpha_{lo})$ and $\hat{\phi}(\alpha_{hi}) = \phi(\alpha_{hi})$, it can be shown that,

$$\begin{aligned} a_1 &= \phi(\alpha_{lo}) \\ a_2 &= \phi'(\alpha_{lo}) \\ a_3 &= \frac{\phi(\alpha_{hi}) - \phi(\alpha_{lo}) - (\alpha_{hi} - \alpha_{lo})\phi'(\alpha_{lo})}{(\alpha_{hi} - \alpha_{lo})^2}, \end{aligned} \tag{6.20}$$

and thus the minimizer is,

$$\alpha = \alpha_{lo} - \frac{\phi'(\alpha_{lo})}{2a_3}. \tag{6.21}$$

If the quadratic is concave the midpoint between α_{lo} and α_{hi} is chosen. Notice however if the starting value of α is acceptable the algorithm terminate immediately.

Algorithm 6.3 Line search algorithm satisfying both Wolfe conditions

Set $k = 0$, $\alpha_0 = 0$, $\tau = 2$, $c_1 = 10^{-6}$, $c_2 = 0.9$ and $LS_{max} = 10$
while $k < LS_{max}$ **do**
  if $\phi(\alpha)_k > \phi(0) + c_1\phi'(0)\alpha$ **then**
    $\alpha =$ `reduce`(α_{k-1}, α_k) and `stop`
  end if
  if $|\phi'(\alpha)_k| \leq -c_2\phi'(0)$ **then**
    $\alpha = \alpha_k$ and `stop`
  end if
  if $\phi'(\alpha)_k \geq 0$ **then**
    $\alpha =$ `reduce`(α_k, α_{k-1}) and `stop`
  end if
  $\alpha_{k+1} = \tau\alpha_k$
  $k = k + 1$
end while

6.4 Solving the Linear Problem

As seen above all algorithm for least-square optimization revolves around solving a linear system of equation. An extensive number of algorithms for solving this problem directly exist. However solving 3D or even 2D image registration problems the size of the matrices and vectors involved prohibits the use of direct solvers. For instance a registration problem involving a $512 \times 512 \times 140$ image with $200 \times 200 \times 100$ basis functions, the image will contain $36,700,160$ voxels, the matrix $\boldsymbol{Q}$ will have the size $110,100,480 \times 12,000,000$ and the vector $\boldsymbol{w}$ will have $12,000,000$ elements. Even the sparsity of $\boldsymbol{Q}$ will not make solving this problem directly possible. The sections below presents tricks to make the solution attainable using properties of the basis functions and iterative schemes for solving the linear systems.

Algorithm 6.4 Reduce subfunction

Call by $\alpha = \texttt{reduce}(\alpha_{lo}, \alpha_{hi})$
loop
  $D = \alpha_{hi} - \alpha lo$
  $a_3 = (\phi(\alpha_{hi}) - \phi(\alpha_{lo}) - D\phi'(\alpha_{lo}))/D^2$
  if $a_3 > 0$ **then**
    $\alpha = \alpha_{lo} - \phi'(\alpha_{lo})/2a_3$
  else
    $\alpha = (\alpha_{lo} + \alpha_{hi})/2$
  end if
  if $\phi\alpha \leq \phi(0) + c_1\alpha * \phi'(0)$ (and) $|\phi'\alpha| \leq -c_2\phi'(0)$ **then**
    $\alpha_{hi} = \alpha$ and `stop`
    return α
  end if
  if $\phi(\alpha) < \phi(0) + c_1\alpha\phi'(0)$ **then**
    $\alpha_l o = \alpha$
  else
    $\alpha_h i = \alpha$
  end if
end loop

6.4.1 The Kronecker Product

The transformations based on separable basis functions can be represented as a Kronecker product. Repeated below is equation 5.22,

$$\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{I}_3 \otimes \boldsymbol{Q}\boldsymbol{w} = \boldsymbol{X} + \boldsymbol{I}_3 \otimes \boldsymbol{Q}_3 \otimes \boldsymbol{Q}_2 \otimes \boldsymbol{Q}_1\boldsymbol{w}, \tag{6.22}$$

which is not only applicable for B-spline basis functions but any separable kernel. The Kronecker product has some nice properties enabling the evaluation of the transformation $\boldsymbol{Y}$.

The iterative schemes described all revolves around matrix-vector products, thus the most important property is [129],

$$\boldsymbol{Q}_2 \otimes \boldsymbol{Q}_1\boldsymbol{w} = \mathrm{vec}(\boldsymbol{Q}_1\boldsymbol{W}\boldsymbol{Q}_2^\top), \tag{6.23}$$

where vec is an operator turning a matrix into a vector. Thus, $\boldsymbol{w} = \mathrm{vec}(\boldsymbol{W})$ which denotes that $\boldsymbol{w}$ is a $nm \times 1$ vector obtained by stacking the columns of $\boldsymbol{W}$. The reverse is $\boldsymbol{W} = \mathrm{reshape}(\boldsymbol{w}, n, m)$ splitting the vector $\boldsymbol{w}$ into m columns of $\boldsymbol{W}$.

In 3D the property also holds however it is more involved to calculate. As in the 2D case each matrix $\boldsymbol{Q}^d$ must operate on the corresponding dth dimension [19].

A method for calculating the product is depicted in algorithm 6.5. The algorithm operates with a 3D dimensional tensor $\boldsymbol{W}$ which is made by reshaping a vector $W_{i,j,k} = w_{i+m_1 j+m_2 k}$. Furthermore two operations has to be defined. First an operation analogous to matrix transpose, a cyclic rearrangement so the last dimension becomes the first, i.e. $\mathbb{R}^{m_1 \times m_2 \times m_3} \rightarrow \mathbb{R}^{m_3 \times m_1 \times m_2}$. Secondly a multiply operation similar to matrix-matrix multiplication but operating on each submatrix along the third dimension of the tensor. This enables the mul-

Algorithm 6.5 3D Tensor-Vector Product: $\boldsymbol{v} = \boldsymbol{Q}_3 \otimes \boldsymbol{Q}_2 \otimes \boldsymbol{Q}_1 \boldsymbol{w}$

Require: $\boldsymbol{Q}_1 \in \mathbb{R}^{n_1 \times m_1}$, $\boldsymbol{Q}_2 \in \mathbb{R}^{n_2 \times m_2}$ and $\boldsymbol{Q}_3 \in \mathbb{R}^{n_3 \times m_1}$, $\boldsymbol{w} \in \mathbb{R}^{m_1 m_2 m_3 \times 1}$
Ensure: $\boldsymbol{v} \in \mathbb{R}^{n_1 n_2 n_3 \times 1}$
1: $\boldsymbol{W} = \text{reshape}(\boldsymbol{w}, m_1, m_2, m_3)$
2: **for** $d = 3$ downto 1 **do**
3: $\quad \boldsymbol{W} = \text{transpose}(\boldsymbol{W})$
4: $\quad \boldsymbol{W} = \text{multiply}(\boldsymbol{Q}_d, \boldsymbol{W})$
5: **end for**
6: **return** $\boldsymbol{v} = \text{vec}(\boldsymbol{W})$

tiplication of a matrix formed by Kronecker product with a vector without calculating the Kronecker product explicitly.

6.4.2 Conjugated Gradients for Least Squares

The core of the Gauss-Newton algorithm is solving the normal equations in equation 6.11. As indicated in the previous section the size of the problem prohibits the explicit formation of the matrices involved and thus an iterative scheme is needed for solving the system. Iterative methods for solving linear problems is an active research area with dozens of algorithms [9, 110]. However, in this thesis the focus is on the conjugate gradients (CG) method [63] which is capable of solving symmetric positive definite systems.

To simplify the equations in the following equation 6.11 is rewritten as,

$$\boldsymbol{J}^\top \boldsymbol{J} \boldsymbol{s} = -\boldsymbol{J}^\top \boldsymbol{b}, \tag{6.24}$$

where $\boldsymbol{J} = \boldsymbol{J}_k$, $\boldsymbol{s} = \boldsymbol{s}_k$ and $\boldsymbol{b} = T(\boldsymbol{Y}) - R(\boldsymbol{X})$. The conjugate gradient method solves the linear problem by minimizing an augmented version of the linear problem,

$$\phi(\boldsymbol{s}) = \frac{1}{2} \boldsymbol{s}^\top \boldsymbol{J}^\top \boldsymbol{J} \boldsymbol{s} + (\boldsymbol{J}^\top \boldsymbol{b})^\top \boldsymbol{s}, \tag{6.25}$$

which has the relation $\boldsymbol{r}_i = \nabla \phi(\boldsymbol{s}_i) = \boldsymbol{J}^\top \boldsymbol{J} \boldsymbol{s}_i + \boldsymbol{J}^\top \boldsymbol{b}$ to equation 6.24. The CG algorithm solves the problem by generating a series of conjugate search directions $\boldsymbol{p}_i$ and converges to the solution by repeatedly solving a 1D minimization

problem along each search direction to gain the next iterate. The solution of the quadratic in equation 10.3 along the direction $\boldsymbol{p}_i$ is given by,

$$\alpha_i = -\frac{\boldsymbol{r}_i^\top \boldsymbol{p}_i}{\boldsymbol{p}_i^\top \boldsymbol{J}^\top \boldsymbol{J} \boldsymbol{p}_i}, \tag{6.26}$$

yielding updates similar to a line search method,

$$\boldsymbol{s}_{i+1} = \boldsymbol{s}_i + \alpha_i \boldsymbol{p}_i. \tag{6.27}$$

Multiplying equation 6.27 with $\boldsymbol{J}^\top \boldsymbol{J}$ and adding $\boldsymbol{J}^\top \boldsymbol{b}$ on both sides yields,

$$\boldsymbol{r}_{i+1} = \boldsymbol{r}_i + \alpha_i \boldsymbol{J}^\top \boldsymbol{J} \boldsymbol{p}_i. \tag{6.28}$$

The next conjugate search directions $\boldsymbol{p}_{i+1}$ is generated using only the previous direction $\boldsymbol{p}_i$,

$$\boldsymbol{p}_{i+1} = -\boldsymbol{r}_i + \beta_i \boldsymbol{p}_i, \tag{6.29}$$

where the scaler β_k can be shown to be [93],

$$\beta_i = \frac{\boldsymbol{r}_{i+1}^\top \boldsymbol{J}^\top \boldsymbol{J} \boldsymbol{p}_i}{\boldsymbol{p}_i^\top \boldsymbol{J}^\top \boldsymbol{J} \boldsymbol{p}_i}. \tag{6.30}$$

Equation 6.29 and the following properties of conjugate directions,

$$\begin{aligned} \boldsymbol{r}_i^\top \boldsymbol{p}_j &= 0 \quad \text{for } j = 0, 1, \ldots, i-1 \\ \boldsymbol{r}_i^\top \boldsymbol{r}_j &= 0 \quad \text{for } j = 0, 1, \ldots, i-1, \end{aligned} \tag{6.31}$$

simplifies equations 6.26 and 6.30 to,

$$\alpha_i = \frac{\boldsymbol{r}_i^\top \boldsymbol{r}_i}{\boldsymbol{p}_i^\top \boldsymbol{J}^\top \boldsymbol{J} \boldsymbol{p}_i}, \tag{6.32}$$

$$\beta_i = \frac{\boldsymbol{r}_{i+1}^\top \boldsymbol{r}_{i+1}}{\boldsymbol{r}_i^\top \boldsymbol{r}_i} \tag{6.33}$$

The condition number of the matrix $\boldsymbol{J}^\top \boldsymbol{J}$ is the square of the condition number of $\boldsymbol{J}$ and this has an impact on the convergence rate of CG [9]. Fortunately the products involving $\boldsymbol{J}^\top \boldsymbol{J}$ can be split in two parts. First the residuals can be phrased as,

$$\boldsymbol{r}_i = \boldsymbol{J}^\top (\boldsymbol{J} \boldsymbol{x}_i + \boldsymbol{b}) = \boldsymbol{J}^\top \boldsymbol{z}_i, \tag{6.34}$$

which leads to the residual update,

$$\begin{aligned} \boldsymbol{z}_{i+1} &= \boldsymbol{z}_i + \alpha_i \boldsymbol{J} \boldsymbol{p}_i \\ \boldsymbol{r}_{i+1} &= \boldsymbol{J}^\top \boldsymbol{z}_{i+1}. \end{aligned} \tag{6.35}$$

Introducing a auxiliary vector $\boldsymbol{q}_i = \boldsymbol{J}\boldsymbol{p}_i$ equation 6.32 simplifies to

$$\alpha_i = \frac{\boldsymbol{r}_i^\top \boldsymbol{r}_i}{\boldsymbol{q}_i^\top \boldsymbol{q}_i} \tag{6.36}$$

Now we have assembled the building block of the conjugated gradients algorithm. A listing can be found in algorithm 6.6.

Algorithm 6.6 CG for Least-squares: $\boldsymbol{J}^\top \boldsymbol{J}\boldsymbol{x} = \boldsymbol{J}^\top \boldsymbol{b}$

while NOT STOP **do**
  Set $\boldsymbol{x}_0 = \boldsymbol{0}$, $\boldsymbol{z}_0 = \boldsymbol{J}\boldsymbol{x}_0 - \boldsymbol{b}$, $\boldsymbol{r}_0 = \boldsymbol{J}^\top \boldsymbol{z}_0$, $\boldsymbol{p}_0 = -\boldsymbol{z}_0$
  $\boldsymbol{q}_i = \boldsymbol{J}\boldsymbol{p}_i$
  $\alpha_i = \frac{\boldsymbol{r}_i^\top \boldsymbol{r}_i}{\boldsymbol{q}_i^\top \boldsymbol{q}_i}$
  $\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha_i \boldsymbol{p}_i$
  $\boldsymbol{z}_{i+1} = \boldsymbol{z}_i + \alpha_i \boldsymbol{q}_i$
  $\boldsymbol{r}_{i+1} = \boldsymbol{J}^\top \boldsymbol{z}_{i+1}$
  $\beta_i = \frac{\boldsymbol{r}_{i+1}^\top \boldsymbol{r}_{i+1}}{\boldsymbol{r}_i^\top \boldsymbol{r}_i}$
  $\boldsymbol{p}_{i+1} = -\boldsymbol{r}_i + \beta_i \boldsymbol{p}_i$
end while

6.5 Regularization

Image registration by simply minimizing equation 6.2 is not usable in practise. There is two reasons for this. One is that the problem is very ill-posed in that the solution is very dependent on the noise in the image data [87]. The second reason is there is nothing in the optimization that ensures the optimal transformation is reasonable [53], e.g. if the deformation field is folded. The remedy of this is to use regularizing which act both as a smoother reducing the ill-posedness of the problem but also ensuring the right properties of the resulting transformation.

6.5.1 Implicit Regularization

Image registration based on transformations parameterized using basis functions contains an implicit regularization. The transformations allowed are restricted to the space spanned by the parameters. An example is the affine transformation discussed earlier where only shearing, rotation and translation are allowed.

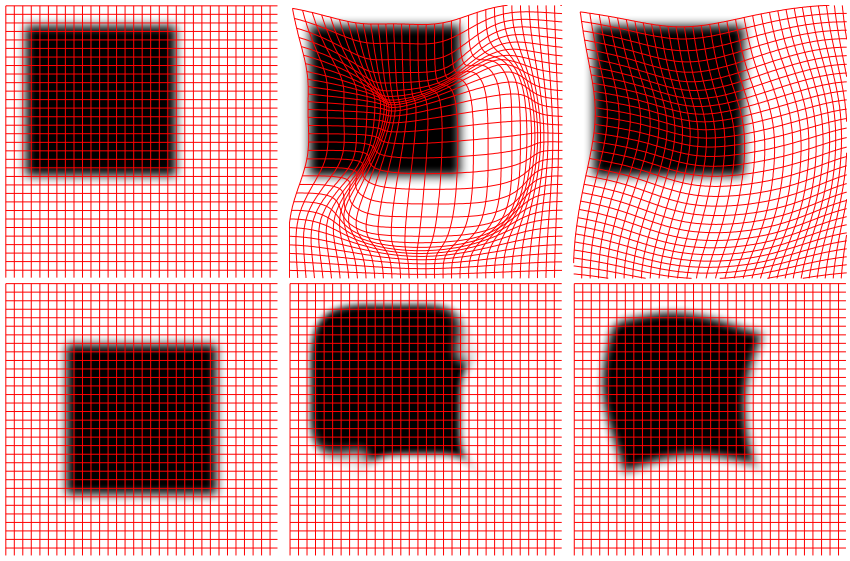


Figure 6.1: Two stages of an image registration. The first column depicts the reference(top) and the template. The next columns depict the registration results with a high number and low number of basis functions respectively. The top row is the reference image overlain the transformation grid. The bottom row is the sampled version of the template image.

In the non-linear case the amount of regularization is determined by the number of basis-functions used. Figure 6.1 shows two stages of an image registration with two levels of basis functions. As seen in figure 6.1 the grid parameterized by a high number of functions is much more irregular than that of figure 6.1 with a lower number.

However the number of basis functions also determines the size of structures that can be registered. Many basis functions might be needed in order to capture fine structures. To overcome this while still maintaining smooth fields a coarse-to-fine approach is used [30, 111, 136]. Here the registration is done by composing a set of transformations with an increasing number of basis functions.

6.5.2 Explicit Regularization

Regularization can also be achieved by adding a penalty term S in 6.2,

$$D(\boldsymbol{w}) = \frac{h}{2}\|T(\boldsymbol{w}) - R\|^2 + \lambda S(\boldsymbol{w}). \tag{6.37}$$

The exact form of the regularizing term depends on the application. One of the most widely used regularizers is Tikhonov Regularization [125]. The formulation of the cost function in equation 6.37 is then,

$$D(\boldsymbol{w}) = \frac{h}{2}\|T(\boldsymbol{w}) - R\|^2 + \lambda\|\boldsymbol{B}\boldsymbol{w}\|^2. \tag{6.38}$$

Typical choices of the matrix $\boldsymbol{B}$ are the first or second derivative operator,

Other types of regularizers tries to induce physical properties into the smoothness requirements of the resulting displacement field. A group of such regularizers are elastic [17], fluid [15, 16, 22, 45], diffusion [41, 53] and many more [87]. Others regularize on the bending energy [13], or add volume preserving constraints [54, 103].

In this work a regularizer based on the diffusion equation is used and thus the regularizing term is [87],

$$S(\boldsymbol{U}) = \|\nabla\boldsymbol{U}\|^2, \tag{6.39}$$

and since $\boldsymbol{U} = \boldsymbol{Q}\boldsymbol{w}$ this can be phrased as in equation 6.38 in terms on the parameters,

$$S(\boldsymbol{w}) = \|\nabla\boldsymbol{Q}\boldsymbol{w}\|^2, \tag{6.40}$$

where $\nabla\boldsymbol{Q}$ is plays the part of the regularizing matrix $\boldsymbol{B}$ from equation 6.38.

The value of the regularization parameter λ can be estimated by plotting an L-curve[58], using generalized cross-validation or by visual inspection [53].

6.5.3 Solving the Augmented Problem

As seen from equation 6.37 the regularizer is an added term to the cost function.

$$\left(\boldsymbol{J}_k^\top\boldsymbol{J}_k + \lambda\boldsymbol{B}^\top\boldsymbol{B}\right)\boldsymbol{s}_k = -\left(\boldsymbol{J}_k + \lambda\boldsymbol{B}^\top\boldsymbol{B}\right)^\top\left(T\left(\boldsymbol{Y}\right) - R\left(\boldsymbol{X}\right)\right), \tag{6.41}$$

where the augmented Jacobian is,

$$\boldsymbol{J}_k^\lambda = \boldsymbol{J}_k + \lambda\boldsymbol{B}^\top\boldsymbol{B}. \tag{6.42}$$

In the case of the diffusion regularizer from equation 6.40, the matrix $\nabla\boldsymbol{Q}$ is defined as [5, 87],

$$\begin{aligned}\nabla\boldsymbol{Q} &= \left(\nabla\boldsymbol{Q}_3^\top\nabla\boldsymbol{Q}_3\right)\otimes\left(\boldsymbol{Q}_2^\top\boldsymbol{Q}_2\right)\otimes\left(\boldsymbol{Q}_1^\top\boldsymbol{Q}_1\right)\\ &+ \left(\boldsymbol{Q}_3^\top\boldsymbol{Q}_3\right)\otimes\left(\nabla\boldsymbol{Q}_2^\top\nabla\boldsymbol{Q}_2\right)\otimes\left(\boldsymbol{Q}_1^\top\boldsymbol{Q}_1\right)\\ &+ \left(\boldsymbol{Q}_3^\top\boldsymbol{Q}_3\right)\otimes\left(\boldsymbol{Q}_2^\top\boldsymbol{Q}_2\right)\otimes\left(\nabla\boldsymbol{Q}_1^\top\nabla\boldsymbol{Q}_1\right),\end{aligned} \tag{6.43}$$

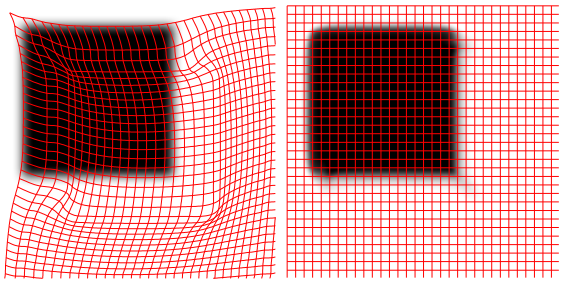


Figure 6.2: Reference and resulting grid $\boldsymbol{y}$ and resulting $T(\boldsymbol{y})$using a diffusion regularizer for the same problem as in figure 6.1.

where $\nabla \boldsymbol{Q}_d$ indicates the matrix of first derivatives of the basis functions in the dth dimension. This augmented problem is also quite large but using Kronecker products makes the use of algorithm 6.6 applicable. Figure 6.2 depicts a registration of the same squares as in figure 6.1 but using a diffusion regularizer.

6.6 Parametric Image Registration Algorithm

As explained above image registration involves similarity measure, interpolation, image transformation, optimization, and regularization. Now all the pieces of an image registration algorithm are present.

As any optimization problem registration is prone to converge to a local minima. To overcome this a multi-resolution, or pyramid, approach is used [1, 107]. Figure 6.3 depicts an image pyramid obtained by successively downsampling the original image. This is a widely used method in image processing and is adopted in many image registration algorithms found in the literature [32, 53, 111, 136]. Basically it is a coarse to fine approach, using a series of downsampled versions of the reference and template. Registration is done on the coarse level propagating the result onto the level above. This has several benefits such as regularization, reduces the number of local minima, provides a good starting guess for the optimization on the next level and reduces computing time.

The algorithm for non-linear registration presented in this thesis consist of two steps; a preregistration step using affine transformations and a non-linear step using the B-spline basis function method. Both uses the multi-resolution method. Listing 6.7 sketches an algorithm for multi-resolution image registration. The propagation step in line 5 of algorithm 6.7 differs between the affine step and the non-linear step. Due to the use of physical coordinates the param-

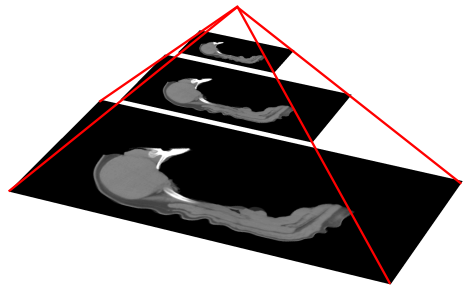


Figure 6.3: The image pyramid is made by downsampling the image number of times. The image can form a pyramid, hence the name.

Algorithm 6.7 Multilevel registration algorithm

1: Initialize $\boldsymbol{w}_0$,
2: **for** $l = l_{min}$ to l_{max} **do**
3: $R_l \leftarrow R$ and $T_l \leftarrow T$ transfer images to level l
4: $\boldsymbol{w}_l \leftarrow$ solve equation 6.2 or equation 6.37 using algorithm 6.1
5: $\boldsymbol{w}_{l+1} \leftarrow \boldsymbol{w}_l$ propagate parameters to next level
6: **end for**

eters in the affine algorithm map directly to the next level, thus $w_{k+1} = w_k$. However in the non-linear algorithm this is not the case.

The non-linear algorithm utilizes an hierarchical approach [79, 111, 136], similar to the multi-resolution method. Here the algorithm starts out using a coarse set of B-spline basic functions on the first level, and refines when a new level is reached and so on. For example an initial configuration of $10 \times 10 \times 10$ basis functions might be refined to a $20 \times 20 \times 20$, ending up with $100 \times 100 \times 100$ on the finest level. Thus there is no direct mapping of parameters from level to level. However an approximate solution can be found. The starting deformation grid $\boldsymbol{U}_{l+1} = \boldsymbol{Q}_{l+1}\boldsymbol{w}_{l+1}$ on a finer level should be more or less equal to the resulting grid $\boldsymbol{U}_l = \boldsymbol{Q}_l\boldsymbol{w}$ on a lower level. However the lower level grid must be prolongated so it has similar size as the fine level. This is done by resampling the basis functions corresponding to $\boldsymbol{Q}_l$ and obtaining $\boldsymbol{Q}_l^\star$. Thus the new set of parameters $\boldsymbol{w}_{l+1}$ can be found by minimizing $\|\boldsymbol{Q}_l^\star\boldsymbol{w}_l - \boldsymbol{Q}_{l+1}\boldsymbol{w}_{l+1}\|^2$ which has the least squares solution

$$\boldsymbol{w}_{l+1} = \left(\boldsymbol{Q}_{l+1}^\top\boldsymbol{Q}_{l+1}\right)^{-1}\boldsymbol{Q}_{l+1}\boldsymbol{Q}_{l+1}^\star\boldsymbol{w}_l \tag{6.44}$$

An algorithm is listed in algorithm 6.8 mapping the parameters by using the least squares approach.

Algorithm 6.8 Propagation of basis-function parameters

1: obtain a finer grid $\boldsymbol{X}_{l+1}$
2: obtain the coarser $\boldsymbol{Q}_l^\star$ using $\boldsymbol{X}_{l+1}$
3: obtain the finer $\boldsymbol{Q}_{l+1}$ using $\boldsymbol{X}_{l+1}$
4: $\boldsymbol{w}_{l+1} \leftarrow \left(\boldsymbol{Q}_{l+1}^\top\boldsymbol{Q}_{l+1}\right)^{-1}\boldsymbol{Q}_{l+1}\boldsymbol{Q}_{l+1}^\star\boldsymbol{w}_l$

6.7 Evaluation of Image Registration

The usefulness of an image registration between two subject is not given by the final value of the similarity metric. A number of other factors should be quantified in order to evaluate the performance [52, 86]. However validation of an image registration is difficult since the ground truth rarely is available. Often the validation is done by measuring the Euclidean distance between anatomical landmarks placed by experts. Recently a project named "Non-Rigid Image Registration Evaluation Project (NIREP)" [23] has been launched. The goal of the project is to develop a standard set of metrics to asses the performance of registration algorithms.

The performance of an registration algorithm must also reflect the application. In the case of mapping landmarks from a reference onto a template the Euclidean distance of landmark is applicable. Basically, the metric is a sum over L Euclidean distances,

$$\mathcal{D}_{lm} = \frac{1}{L}\sum_{i=1}^{L} \|\phi(\boldsymbol{x}_{ri}) - \boldsymbol{x}_{ti}\| = \frac{1}{L}\sum_{i=1}^{L} \|\boldsymbol{x}_{ri} + u(\boldsymbol{x}_{ri}) - \boldsymbol{x}_{ti}\|, \tag{6.45}$$

where in $u(\boldsymbol{x}_r)$ is the displacement in $\boldsymbol{x}_r$ and is found by linearly interpolating the resulting displacement field $\boldsymbol{U}$. See section 5.2. Another measure suited for registration applications involving atlases is the Intensity Variance Metric [23]. The underlying idea is that when a series of images is registered to the same reference the average of the transformed images should be sharp. Sharpness can be measured by computing the variance of the transformed images. The variance image is computed as,

$$I_{iv} = \frac{1}{L}\sum_{1}^{L} (T_i(\boldsymbol{y}_i) - A(\boldsymbol{x}))^2, \tag{6.46}$$

where L is the number of transformed images and $A = \frac{1}{L}\sum_{i=1}^{L} T_i(\boldsymbol{y}_i)$ is the average image.

Part III

Applications

Chapter 7

Virtual Jointing of Pig Carcasses using Image Registration

Martin Vester-Christensen and Lars Bager Christensen and Marchen Hviid and Eli V. Olsen and Rasmus Larsen

Abstract

In the Danish pig slaughtering industry optimizing the use of each pig is of major importance. A large number of different cuts of each carcass is possible. Not all cuts are suited for each carcass due to their obvious anatomical difference. Optimal use is achieved by sorting the carcasses into groups. During the slaughtering process a set of measurements of lean meat percentage and weight is made on each carcass. These measurements form the parameters used in the sorting. Thus lowering the anatomical variation within each group substantially. It is important to know the relation between the quality of a cut, in this case the yield, and the measured parameters in order to optimize the use of the carcasses. This calibration is done by manually cutting carcasses and estimating quality of the individual cuts. This is very labor intensive and as such only a limited number of carcasses is used.

In this paper a method for predicting the yield of two different products is proposed. The anatomical relation of each carcass and a reference is

established using image registration of Computed Tomography(CT) scans of the carcasses. Using this relation a series of virtual cuts defined by anatomical landmarks can be propagated onto each carcass. The yields of these virtual cuts are calculated and related to the lean meat percentage and the carcass weight.

7.1 Introduction

The pig industry is one of the largest and most important industries in the Danish economy with an export value of approximately 28 billion DKK [113]. As a consequence of increasing competition from slaughterhouses in countries with lower wages, optimization of all aspects of the slaughtering process is needed. The inhomogeneous nature of the pigs with obvious differences in weight, size, etc. makes optimal use difficult. The slaughterhouses tries to remedy this by sorting the carcasses into groups of similar properties. All carcasses in a group is then cut into the same product. The sorting criteria and the sorting accuracy has substantial influence on the economy of the slaughterhouses [75].

The carcasses can be used in a variety of different end-products. However yield, costs and prices determine the suitability of a carcass being cut as a particular product. Yield of a product is defined as the fraction of meat versus the combined weight of meat and fat in the product.

In order to ensure optimal use of the carcasses, it is important to have knowledge of how well the carcasses fits a given cut. The slaughterhouses uses mathematical models to predict the quality of particular cuts. These models are calibrated by manually cutting a set of pigs into a given product. The quality, e.g. yield, of the cuts is then measured. This process is very labor intensive which constrains the number of carcasses used in the calibration. Furthermore a carcass can only be cut once, and thus its compatibility with other cuts cannot be established.

Recently research in the use of Computed Tomography(CT) as a tool in the Danish slaughterhouses has been undertaken. This paper is a preliminary study in the use of CT as a tool for predicting the quality of specific uses of a pig carcass in the Danish slaughterhouses.

Image registration is used to obtain an anatomical mapping between a reference carcass and a set of template carcasses. To the reference is coupled a set of anatomical landmarks. The use of these landmarks is two-fold; firstly they are used a measure of registration accuracy and secondly a set of virtual cuts is defined relatively to the landmarks. The anatomical mapping obtained from the image registration is used to propagate the virtual cuts onto the set of carcasses. This enables the evaluation of the virtual cuts in all carcasses without the use

of elaborate feature extraction schemes.

Two cuts of the middle part of the carcass are evaluated. A loin product which is produced by cutting very close to the muscle longissimus dorsi. Furthermore an 18cm back product which is produced by making a cut parallel with the rind at a distance of 18cm from the rind edge. Simple models for the prediction of the yield in the loin and the 18cm back product are created. The parameters of the models are the total weight and the lean meat percentage. Two parameters actually used in the slaughterhouses today. The method is evaluated by predicting the yield of 57 carcasses.

7.1.1 Previous work

Earlier work using image analysis to virtually cut pig carcasses is found in [56]. Hansen uses a slice by slice approach to virtually cut the 18cm back with good results. Features are extracted by searching radially from a predefined location in the slice.

In this work the cuts are made on a 3D carcass volume using image registration and the propagation of landmarks as the cutting algorithm. Consequently much of the previous work is to be found in the image registration literature. By far the most dominating field is medical image registration.

Image registration is a very active research area. Especially in the field of medical imaging but also in a variety of other applications. Several methods and applications have been published. Books covering the topic are found in Modersitzki [87] and Hajnal *et al.* [55]. Thorough surveys of the many algorithms and applications can be found by Zitova and Flusser [137], by Pluim *et al.* [100], by Maintz and Viergever [86], and by Brown [18]. Basically the task is to find the optimal dense field transforming a template image into a reference image. Several approaches exist. One type is algorithms using parameterizations of this transformation field using basis functions [4, 77, 108]. Another uses non-parametric algorithms often based on solving fluid or elastic partial differential equations [22, 87]. Several software packages are available and among them are the popular AIR package [135], the tools NA-MIC [99], the SPM package [5]

Atlas construction is also an active area of research. Most predominately is brain atlases [14, 20, 69, 124] but also other applications [94] exist. The application of atlases most similar to the application in this paper is atlas based segmentation [102, 104, 105]. Here the atlas comes with an annotation segmenting one or several structures in the brain or other object.

Figure 7.1: A half carcass prepared and ready for scanning. Courtesy of Danish Meat Association.

7.2 The Data

The data used are 57 left sides of half pig carcasses scanned using a CT scanner. A half pig carcass is the product of the pig carcass being split along the spine. In most Danish slaughterhouses this is an automated process in which a robot uses a saw to perform the split. After the splitting the head and toes are cut off and the carcass is cooled. The day after slaughtering the left side of the carcass is ready for scanning. A half carcass ready for scan is shown in figure 7.1.

The pigs in the set is chosen to be representative of the Danish pig population ready for slaughter with respect to age, weight and fatness. The experiment is a part of large calibration trial conducted in the spring of 2008. See [131](chapter 9) for a more thorough description of the trial and experiment. The full carcass weight ranges from approximately 60 kg to about 100 kg and they are slaughtered at an approximate age of six months. Figure 7.2 and figure 7.3 depicts the ranges of the weight and lean meat percentage of the data used.

The carcasses are scanned using a refurbished GE HiSpeed CT/i single-slice scanner capable of isotropic voxels. The voxel dimensions are $1 \times 1 \times 10$ mm. An average prepared half carcass is around 130 cm in length which yields a volume of $512 \times 512 \times 130$ voxels.

7.3 Image Registration

Image registration is the task of finding a reasonable transformation such that a transformed version of a template image T becomes similar to a reference image R [53]. Transformation is described using linear and non-linear models and the transforming images is done using interpolation schemes. The optimal

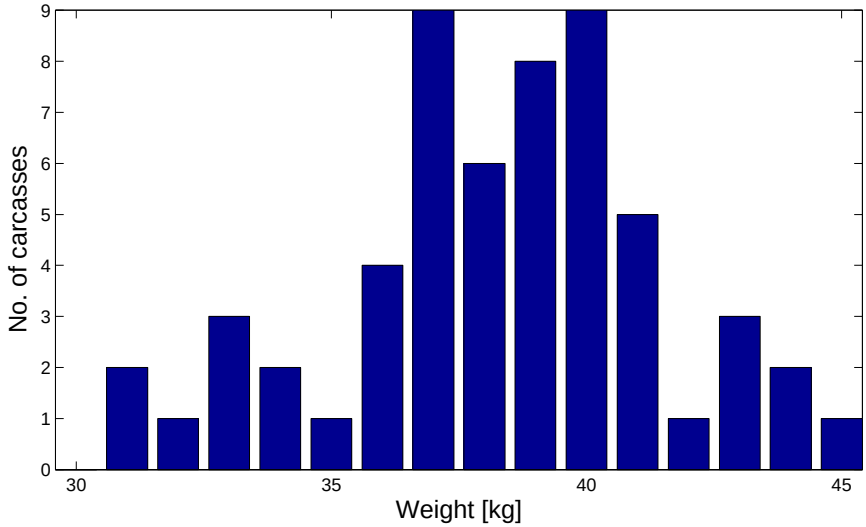


Figure 7.2: Distribution of weight in the data set.

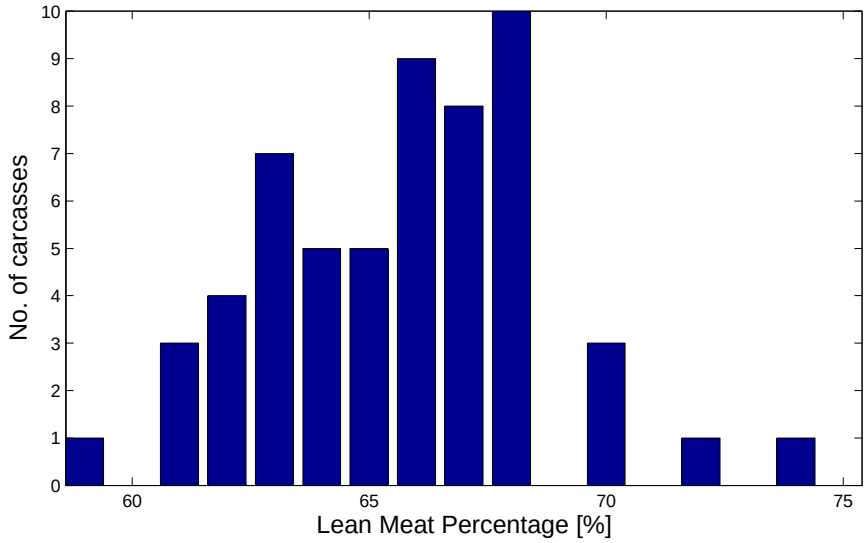


Figure 7.3: Distribution of lean meat percentage in the data set.

transformation is described using some kind of similarity measure and finding it is the task of optimization. Adding a regularization term in the optimization ensures that the transformation is smooth.

7.3.1 Linear and Non-Linear Transformation Models

In this paper an image is seen as a continuous function defined on some domain. Sampling the continuous image on a grid of points enables the discretization needed. The sampling is done using an interpolation scheme. Many exist but in this paper a B-spline based scheme is used [120].

A grid is defined as a collection of grid points $\boldsymbol{x}_i = (x_i^1, \ldots, x_i^D)^\top$ for a D-dimensional image. A 3D grid $\boldsymbol{X}$ of size $n_1 \times n_2 \times n_3$ is defined as,

$$\boldsymbol{X} = \left[\boldsymbol{x}^{1\top}\ \boldsymbol{x}^{2\top}\ \boldsymbol{x}^{3\top}\right]^\top, \tag{7.1}$$

where $\boldsymbol{x}^1 = \left(x_1^1 \ldots x_N^1\right)^\top$, $\boldsymbol{x}^2 = \left(x_1^2 \ldots x_N^2\right)^\top$ and $\boldsymbol{x}^3 = \left(x_1^3 \ldots x_N^3\right)^\top$, with $N = n_1 n_2 n_3$.

An image is transformed by transforming the underlying grid and then resampling the image. The transformation can be seen as a sum of an identity part and a deformation part,

$$\begin{aligned} y_i^1 &= x_i^1 + u^1(\boldsymbol{x}_i) \\ y_i^2 &= x_i^2 + u^2(\boldsymbol{x}_i) \\ y_i^3 &= x_i^3 + u^3(\boldsymbol{x}_i), \end{aligned} \tag{7.2}$$

where $u^d(\boldsymbol{x}_i)$ is the displacement of a grid point $\boldsymbol{x}_i$ in the dth dimension.. Thus for each grid point there is a corresponding displacement vector comprising a vector field.

In image registration the images are usually preregistered using linear transformations. In an atlas application it is desirable to remove effects from the image acquisition process from the model. The most flexible linear transformation is the affine transformation, which can model rotation, translation, scaling and shearing. The transformation of the entire grid $\boldsymbol{X}$ can be written as,

$$\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{I}_3 \otimes \boldsymbol{Q}\boldsymbol{w}, \tag{7.3}$$

where $\boldsymbol{I}_3$ is a unit matrix of rank 3. The q_{ij}th element of $\boldsymbol{Q}$ is the evaluation of the jth basis function at the ith position. $\boldsymbol{w}$ is a vector of the parameters for the corresponding basis functions. For the linear transformation the very simple basis functions are,

$$q_1(\boldsymbol{x_i}) = x_i^1, \quad q_2(\boldsymbol{x_i}) = x_i^2, \quad q_3(\boldsymbol{x_i}) = x_i^3, \quad q_4(\boldsymbol{x_i}) = 1. \tag{7.4}$$

A type of transformation capable of modeling non-linear deformations is based on cubic B-splines [108]. As above the transformation can be written as a sum of basis functions,

$$y_i^d = x_i^d + \sum_{j=1}^{p_1}\sum_{k=1}^{p_2}\sum_{l=1}^{p_3} b_j(x_i^1)b_k(x_i^2)b_l(x_i^3)w_{jkl}^d, \tag{7.5}$$

where p_1,p_2 and p_3 are the number of equidistantly placed knots for the splines. Using the tensor product property of B-splines equation 7.5 can be written in similar fashion as equation 7.3,

$$\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{I}_3 \otimes \boldsymbol{Q}_3 \otimes \boldsymbol{Q}_2 \otimes \boldsymbol{Q}_1 \boldsymbol{w}. \tag{7.6}$$

Using a non-recursive definition [121] the following 1D cubic B-splines basis functions can be derived,

$$b_j(x_i) = \begin{cases} (2+x_i)^3 & \text{for } -2 \le x_i < -1 \\ -(3x_i+6)x_i^2 + 4 & \text{for } -1 \le x_i < 0 \\ (3x_i-6)x_i^2 + 4 & \text{for } 0 \le x_i < 1 \\ (2-x_i)^3 & \text{for } 1 \le x_i < 2 \\ 0 & \text{otherwise} \end{cases} \tag{7.7}$$

7.3.2 Similarity and Optimization

The data in this paper consist of CT-scan of pig carcasses, and as such the registration is between two intra-modal images. A similarity measure suited for this situation is the sum-of-squared-differences (SSD) measure. It measures the Euclidean distance between grey values of corresponding voxels,

$$D(\boldsymbol{w}) = \frac{h}{2}\|T(\boldsymbol{w}) - R\|^2, \tag{7.8}$$

where $h = h_1h_2h_3$ is the product of the voxel dimensions. A good choice for optimization of SSD-problems is the Gauss-Newton algorithm. The basic idea is iteratively updating the parameters $\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \boldsymbol{s}_k$ by solving the linear system also known as the normal equations [93],

$$\boldsymbol{J}_k^\top \boldsymbol{J}_k \boldsymbol{s}_k = -\boldsymbol{J}_k^\top \left(T\left(\boldsymbol{Y}\right) - R\left(\boldsymbol{X}\right)\right), \tag{7.9}$$

where $\boldsymbol{J}_k$ is the Jacobian $J(\boldsymbol{w}_k)$. Thus it only involves calculating the residuals and the Jacobian. In this parametric case the Jacobian can be written as,

$$\boldsymbol{J}_k = \boldsymbol{\nabla T}\boldsymbol{Q} = \left[\boldsymbol{\nabla T}^1\ \boldsymbol{\nabla T}^2\ \boldsymbol{\nabla T}^3\right] \boldsymbol{I}_3 \otimes \boldsymbol{Q}_3 \otimes \boldsymbol{Q}_2 \otimes \boldsymbol{Q}_1, \tag{7.10}$$

where each $\boldsymbol{\nabla T}^d$ is a diagonal matrix containing the partial derivatives $\partial T(\boldsymbol{y}_i)/\partial x^d$ of the image T.

Image registration problems can become very large systems. For instance a non-linear registration problem involving a $512 \times 512 \times 140$ image with $200 \times 200 \times 100$ basis functions, the image will contain $36,700,160$ voxels, the matrix $\boldsymbol{Q}$ will have the size $110,100,480 \times 12,000,000$ and the vector $\boldsymbol{w}$ will have $12,000,000$ elements. Even the sparsity of $\boldsymbol{Q}$ will not make solving this problem directly possible. However using properties of Kronecker products enables the calculation of the product $\boldsymbol{Qw}$ without explicitly forming $\boldsymbol{Q}$ [19, 98]. This enables the use of conjugate gradient type algorithms in which only matrix-vector products are needed [39, 90, 93].

7.3.3 Regularization

Regularization is introduced in order to successfully solve the linear system in equation 7.9 and to ensure that the transformation field $\boldsymbol{y}$ is smooth. It is done by augmenting the similarity measure in 7.8 with a regularizer term,

$$D(\boldsymbol{w}) = \frac{h}{2}\|T(\boldsymbol{w}) - R\|^2 + \lambda S(\boldsymbol{w}).,\tag{7.11}$$

where $\boldsymbol{B}$ is a matrix operator implementing the regularization scheme of choice operating on the displacement field $\boldsymbol{u}$.

In this paper a diffusion type regularizer is used [4, 41]. Thus the regularizing term is,

$$S(\boldsymbol{w}) = \|\nabla \boldsymbol{Qw}\|^2.\tag{7.12}$$

7.4 Atlas Construction

The construction of the atlas is very simple. To obtain maximal contrast and to ease the placement of landmarks a representative of the 57 carcasses is chosen to be the atlas. This might introduce a bias towards the atlas carcass, however in this application no statistical inference is made making this less important.

A more elaborate procedure follows the algorithm presented in [51] and [109]. Here all subjects is registered to one chosen subject. The average of all registered subjects is then made the new atlas and the subjects are registered to this. This continues until the atlas stops changing. To remove the bias the atlas is then transformed with the average of the inverse of all transformations.

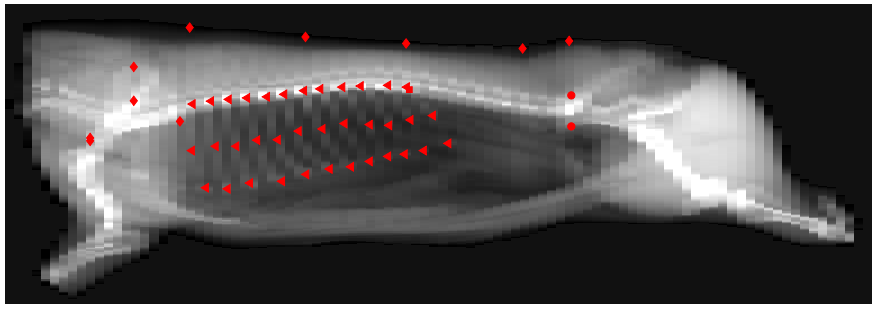


Figure 7.4: A top-down view of the landmarks placed on the virtual carcass.

7.5 Virtual Jointing

In this section the virtual cuts mimicking actual cuts in the Danish slaughterhouses are defined. A diagram of a pork skeleton is depicted in figure 7.5. Many actual cuts made by a butcher or a robot are relative to some reference points located on the skeleton of the carcass and likewise for the virtual cuts.

Several landmarks have been placed in the atlas and an overview of the landmarks in use can be found in figure 7.4. The landmarks are placed as follows,

- 5 around the scapula used to define the fore-end of the carcass.
- 2 around the ilium used to define the start of the legs part of the carcass.
- 12 where the twelve most posterior ribs join the thoracic vertebrae.
- 12 at the left most part of the twelve ribs counted from the posterior end to the anterior.
- 12 at the tip of the twelve most posterior ribs.
- 1 at the tip of the neural spine in the same slice as where the most posterior rib join the thoracic vertebrae. Denoted the $P2$ point.
- 1 at the bottom of the loin in the same slice as the $P2$ point.
- 5 along the rind of the carcass.

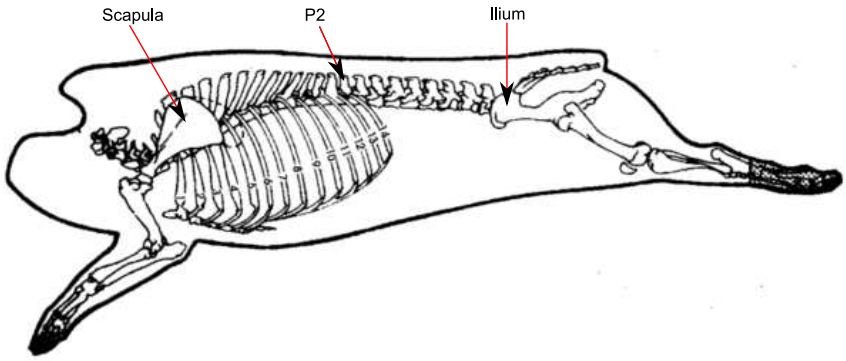


Figure 7.5: A diagram of a pig skeleton [2].

Image registration is used to propagate the landmarks from the atlas onto each target carcass. By linearly interpolating the resulting transformations $\boldsymbol{Y}$ the target location of each landmark can be found.

7.5.1 Splitting into Three

Two sets of cuts are simulated. Common for both sets it the splitting of the carcass into three parts; a front-end, a middle and a leg part.

The fore-end is defined to be from the anterior end of the carcass to the most posterior point on the scapula. The middle is defined to be from the scapula to the middle of the ilium. Finally the legs part of the carcass is from the middle of the ilium to the most posterior end of the carcass. All cuts are made as planes perpendicular to the sagittal plane simulating a straight cut from top to bottom through the carcass.

Figure 7.6 depicts the splitting planes. However when the split is performed on the slaughterline, the spine of the carcass is straightened out. Notice in figure 7.6 that the spine is not straight when the carcass is scanned. To remedy this the splitting plane between the middle and the leg part is defined by a normal vector with the direction from the ilium to the $P2$ point as normal. The splitting plane separating the head part from the rest of the carcass lies in the axial plane. Figure 7.7 depicts the resulting parts of three carcasses after a split into three.

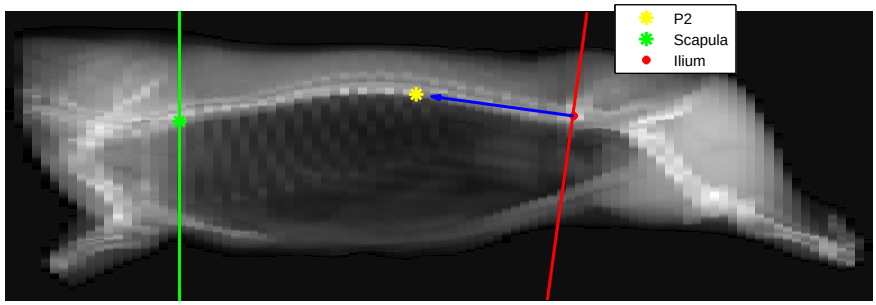


Figure 7.6: A carcass with the landmarks and planes used for splitting the carcass in three parts. The red dot denotes the ilium, the yellow the $P2$ point and the green the posterior end of the scapula. The green line indicates the splitting plane separating the head part from the rest of the carcass. The red line denotes the splitting plane separating the leg part from the carcass.

7.5.2 Splitting into Loin and Belly

The second split is where the two product sets differ. In the first product the middle is cut into a loin part and a belly part. Figure 7.8 shows the loin and belly product. As with the first split the spine of the carcass is straight when the cut is performed. However more or less all carcasses have a curved spine when scanned. To overcome this the cut is made from two planes. When cutting the robot or butcher uses a set of guide points. Here the points are defined relative to the landmarks.

The first point is placed on the most anterior of the annotated ribs. Figure 7.9 depicts the location of the guide point $\boldsymbol{X_r}$. The location is defined as 4cm from the landmark L_{m1} at the spine along the vector V which points along the rib towards the landmark L_{m2} at the middle of the rib.

Figure 7.9 describes the location of the second guide point $\boldsymbol{X}_p$. It is positioned at the most posterior rib, 8cm from the start of the rib at landmark L_{m1}.

The posterior guide point $\boldsymbol{X}_i$ is located 2cm from the landmark defining the leftmost point of the ilium, see figure 7.11.

The guide points span two planes used for the virtual cuts. Figure 7.12 depicts the two planes as lines very the planes cut the midsagittal plane.

The split in the anterior part of the carcass relative to $\boldsymbol{X}_s$ is defined by the plane L_1 colored in red in the figure, while the green line indicates the plane L_2 defining the split in the anterior part.

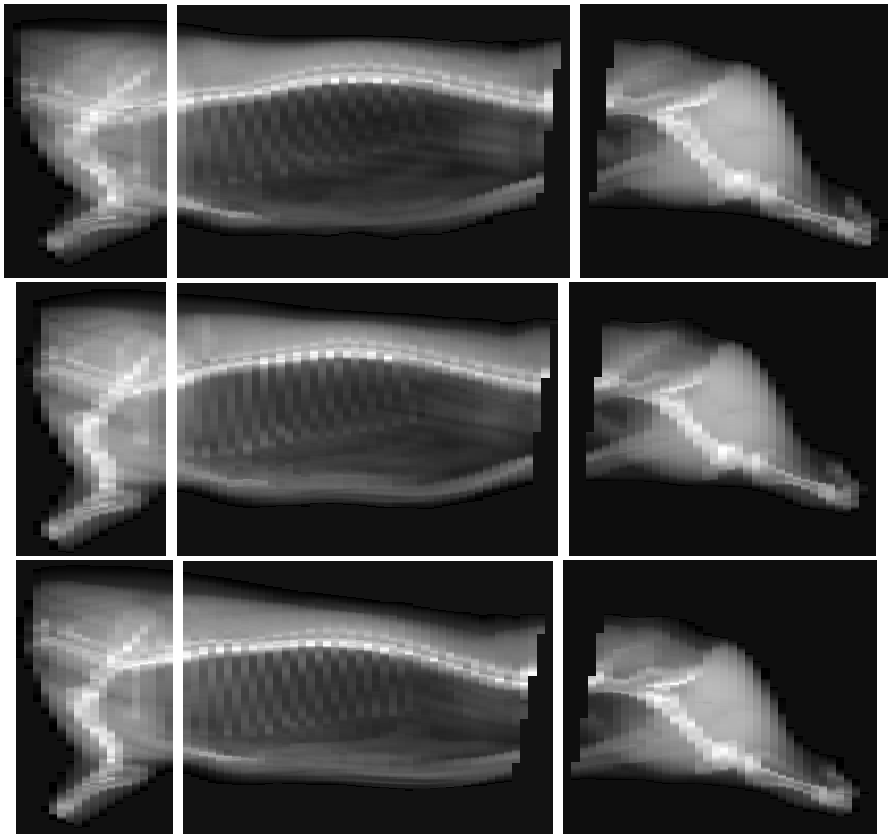


Figure 7.7: The resulting parts after a split into three for three different carcasses.

Figure 7.8: Actual loin(top) and belly product.

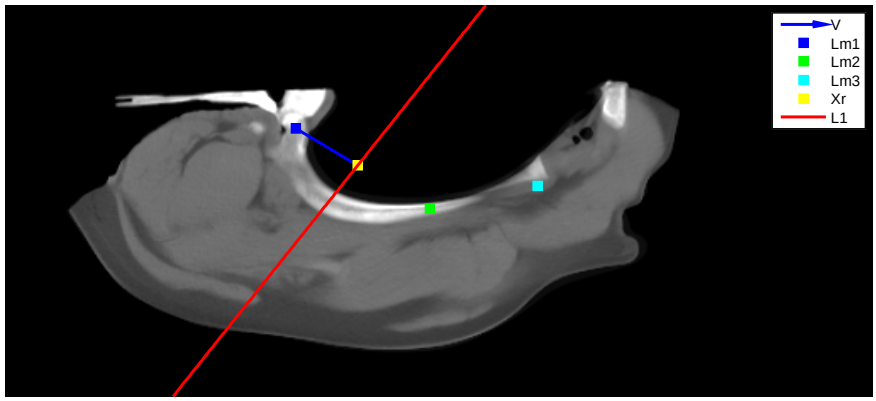


Figure 7.9: The most anterior guide point $\boldsymbol{X_r}$. It is defined to be 4cm from the landmark L_{m1} along the vector V in the direction of the landmark L_{m2}. The red line L_1 indicates the intersection between the plane L_1 and the axial plane containing the guide point $\boldsymbol{X}_r$.

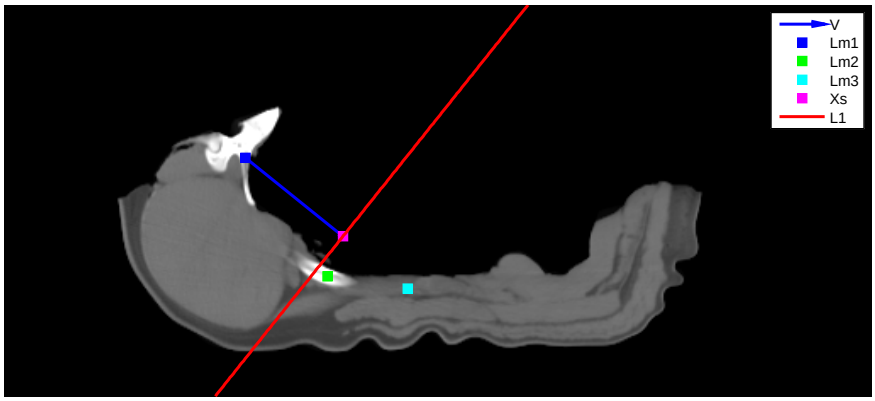


Figure 7.10: The middle guide point $\boldsymbol{X}_s$. It is defined to be 8cm from the landmark L_{m1} along the vector V in the direction of the landmark L_{m3}. The red line L_1 indicates the intersection between the plane L_1 and the axial plane containing the guide point $\boldsymbol{X}_s$.

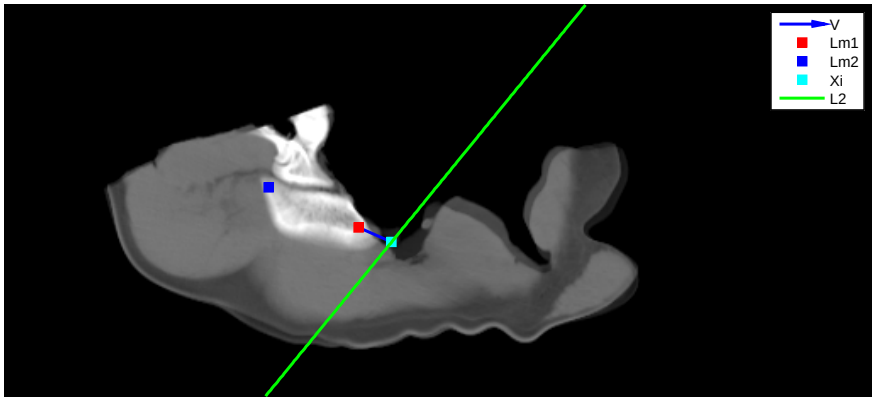


Figure 7.11: The posterior guide point $\boldsymbol{X}_i$. It is defined to be 2cm from the landmark L_{m2} along the vector V which has the direction from L_{m1} to L_{m2}. The green line L_2 indicates the intersection between the plane L_2 and the axial plane containing the guide point $\boldsymbol{X}_i$.

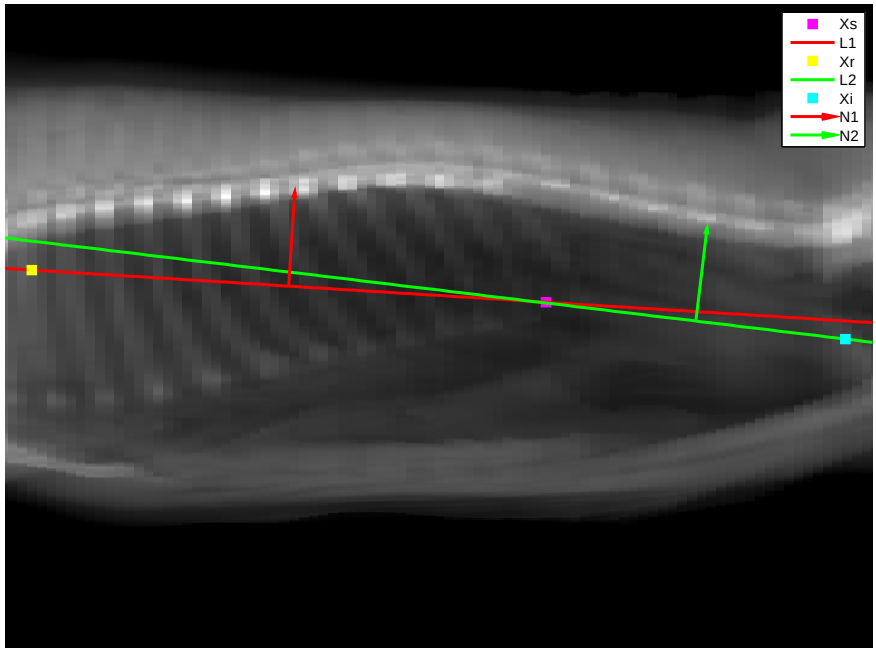


Figure 7.12: The two splitting planes defining the loin part.

On the slaughterline the cuts are made to be perpendicular to the skin of the carcass, thus the cuts are not perpendicular to the sagittal plane. To simulate this the planes are at an angle to the midsagittal plane.

Each plane is defined by a point and a normal vector. A natural point to choose for both planes is the middle guide point $\boldsymbol{X}_s$. The normal vector of the plane L_1 is perpendicular to the vector from $\boldsymbol{X}_r$ to $\boldsymbol{X}_s$, with an angle to the midsagittal plane given by from the vector V seen in figure 7.10.

The normal vector of the plane L_2 is perpendicular to the vector from $\boldsymbol{X}_s$ to $\boldsymbol{X}_i$, and the rotation stems from the same vector V as in L_1.

When the two planes are defined the virtual cuts can be executed by investigating the sign of the inner product between the normal vectors and the coordinate vectors of each voxel. A result of a virtual cut can be seen in figure 7.13 in which three equidistant axial slices of the two products is displayed.

7.5.3 Splitting into Back and Belly

The second product consist of a 18cm back and a belly part. Here the 18cm back is defined as the loin and the part of the belly 18cm from the rind. The five landmarks along the rind is used, and the guide points is just the landmark translated 18cm in the axial plane. The cut is defined by four planes given by the five guide points. The cut is made perpendicular to the sagittal plane so no rotation is needed. Figure 7.14 depicts the rind points, guide points and intersections with the planes and the midsagittal plane. Like above the cut is evaluated by investigating the sign of the inner product between the normal vectors and the coordinate vectors of each voxel. In figure 7.15 the examples of a virtual 18cm back and belly which three equidistant axial slices of the two products is displayed.

7.6 Yield Prediction Models

The models used to predict the yield of a product are simple linear models [59] of the form,

$$g_i = \beta_1 v_i + \beta_2 p_i + \beta_3, \tag{7.13}$$

where g_i is the yield, v_i is the weight of the half carcass, p_i is the lean meat percentage. β_1, β_2 and β_3 are the corresponding parameters. The basis of these models is the ability to estimate the weight of the meat, fat and bone in the cut, and to estimate the lean meat percentage in the whole carcass. The weight and

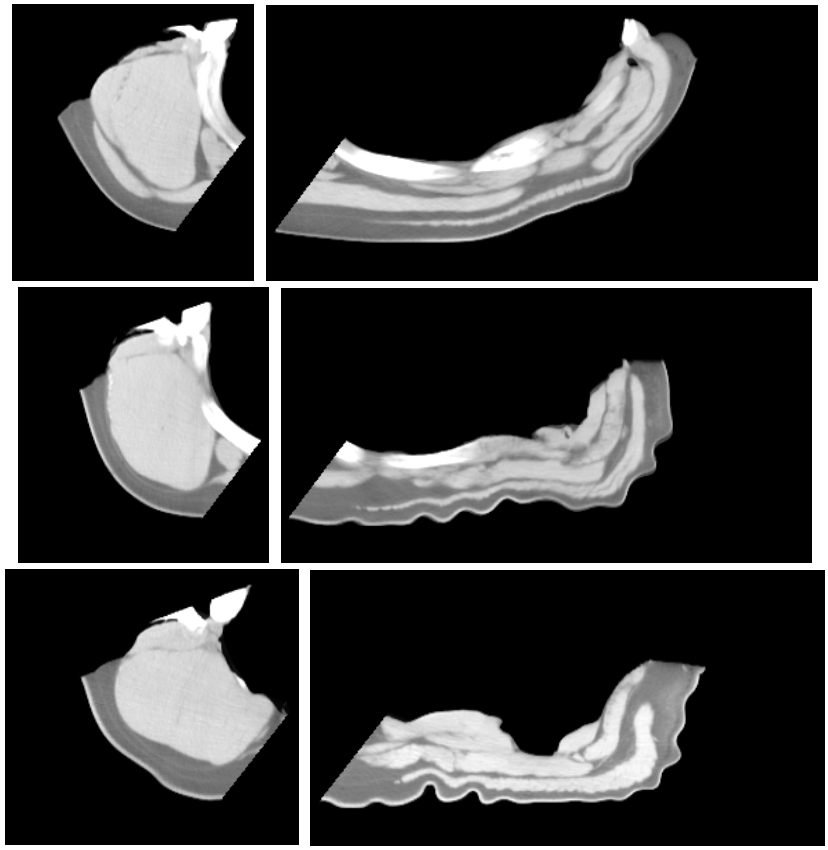


Figure 7.13: The resulting loin(left) and belly parts.

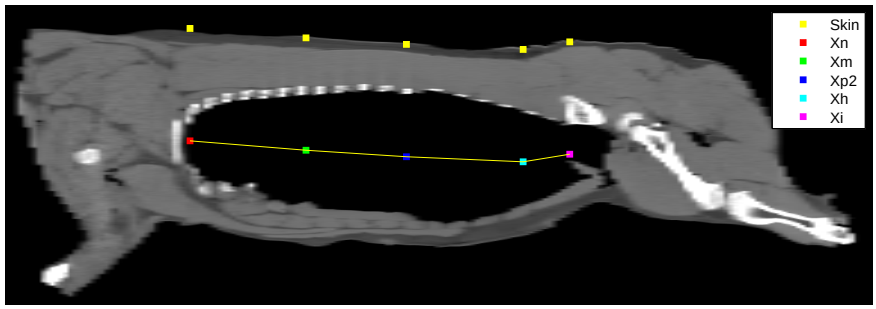


Figure 7.14: Cutting planes defining the 18cm back and belly product.

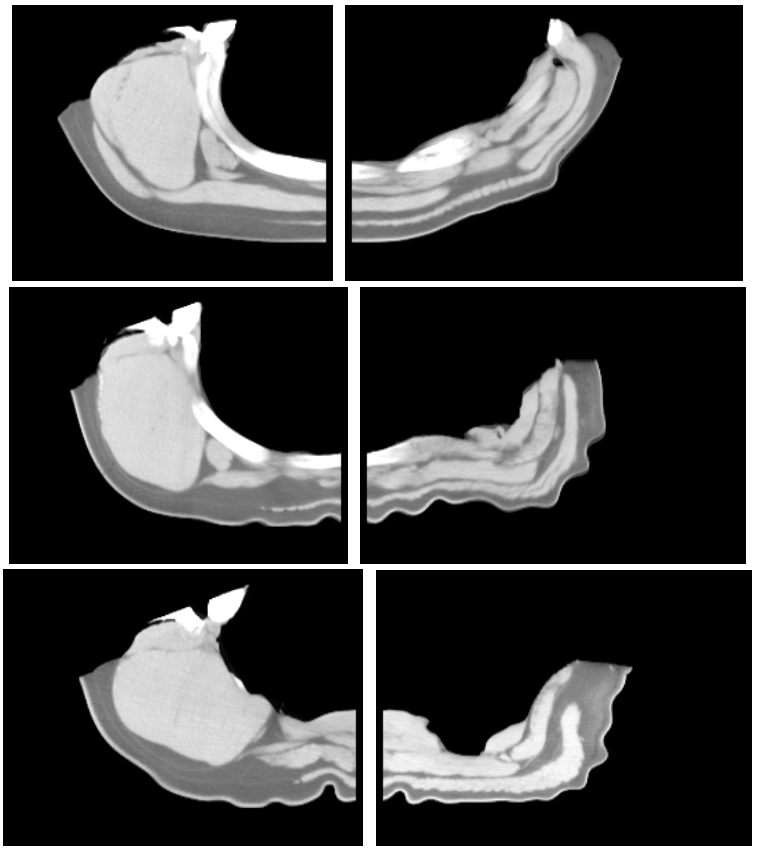


Figure 7.15: The resulting back(left) and belly parts.

the lean meat percentage is estimated from the CT volume data using linear model based on a pixel classification algorithm described in [131](see chapter 9).

7.7 Results

All 57 carcasses are registered to an atlas chosen as a carcass representative of the population. Each carcass volume is resampled to a volume size of $512 \times 256 \times 192$ voxels. For the non-linear part of the registration the basis functions are placed 20, 10 and 5mm apart for the coarse to fine level respectively. Six different registrations are performed on each of the 24 carcasses correspond to six levels of the regularizing parameter λ. The values are $\lambda_i = \left[10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}\right]$ for $i = 1 \ldots 6$.

7.7.1 Choice of Regularization Parameter

In order to determine the optimal value for the regularization parameter two quantitative measures are used.

One quantitative measure is based on landmarks. 24 carcasses was selected and annotated using nine of the landmarks described in the scheme described in section 7.5. The landmarks are the five around the scapula, the two around the ilium, the $P2$ point and the loin point. The landmarks are used to evaluate the registration accuracy by measuring the Euclidean distance between the automatically placed and its corresponding manually placed. Basically, the metric is a sum over 24 Euclidean distances in carcass j,

$$\mathcal{D}_j^{lm} = \frac{1}{9}\sum_{i=1}^{9} \|\phi(\boldsymbol{x}_{ri}) - \boldsymbol{x}_{ti}\|^2 = \frac{1}{9}\sum_{i=1}^{9} \|\boldsymbol{x}_{ri} + u(\boldsymbol{x}_{ri}) - \boldsymbol{x}_t\|^2, \tag{7.14}$$

where $\boldsymbol{x}_{ti}$ is the manually placed landmark i, $\boldsymbol{x}_{ri}$ is the corresponding propagated landmark. $u(\boldsymbol{x}_r)$ is the displacement in $\boldsymbol{x}_r$ and is found by linearly interpolating the resulting displacement field $\boldsymbol{U} = \boldsymbol{Y} - \boldsymbol{X}$. As a score for a particular value of the regularization parameter is the average landmark error,

$$\mathcal{D}_\lambda^{lm} = \frac{1}{24}\sum_{i=1}^{2} 4\mathcal{D}_j^{lm}, \tag{7.15}$$

The left plot in figure 7.16 depicts a the mean and standard deviation of the Euclidean distances.

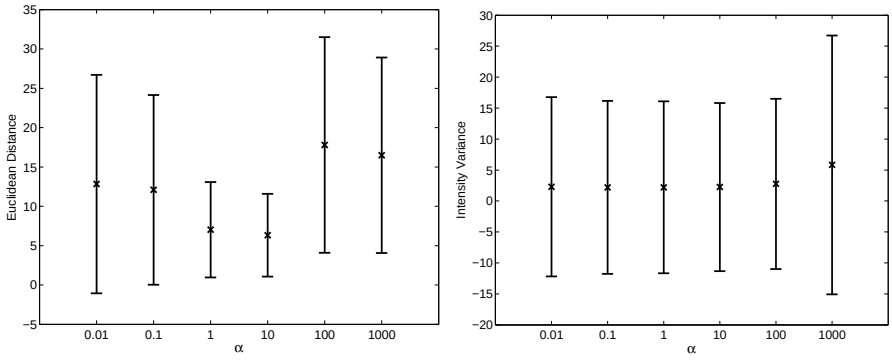


Figure 7.16: Landmark errors(left) and intensity variance values for six levels of the regularization parameter λ. The x denotes the mean and the whiskers denotes the range of one standard deviation.

The second quantitative measure is denoted The Intensity Variance Metric [23]. When a series of images is registered to the same reference the transformed images should be similar. Thus the average of all the transformed image should be sharp. Sharpness can be measured by computing the variance of the transformed images. The variance image is computed as,

$$\mathcal{I}_j^{iv} = \frac{i=1}{L} \sum_{1}^{L} (T_i(\boldsymbol{Y}_i) - A(\boldsymbol{X}))^2, \tag{7.16}$$

where L is the number of transformed images and $A = \frac{1}{L} \sum_{i=1}^{L} T_i(\boldsymbol{Y}_i)$ is the average image. As a score for a particular value of the regularization parameter is the average intensity variance,

$$\mathcal{I}_\lambda^{iv} = \frac{1}{24} \sum_{i=1}^{24} \mathcal{I}_j^{iv}, \tag{7.17}$$

The right plot in figure 7.16 shows the distribution of intensity variances across six levels of regularization.

Table 7.1 recaps the values of figure 7.16. Here it can be seen that the value of the regularization parameter should be either $\lambda = 1$ or $\lambda = 10$ with the latter as the chosen one. Figure 7.17 depicts the midaxial, midsagittal, and midcoronal slices of the atlas with a rendering of the deformation field overlayed.

With this choice of the regularization parameter the remainder of the 57 carcasses are registered to the atlas. The registration including both the affine and non-linear step took on average of 168 minutes per carcass using a Matlab

λ	0.01	0.1	1	10	100	1000
$\mathcal{D}_\lambda^{lm}[mm]$	12.8	12.1	7.0	6.3	17.7	16.4
$\mathcal{I}_\lambda^{iv}$	2.2	2.1	2.1	2.2	2.7	5.8

Table 7.1: The average landmark and intensity variance for six levels of regularization.

	β_1	β_2	β_3
Loin	0.0005 [−0.0005 0.0016]	0.0131 [0.0119 0.0142]	−0.1423 [−0.2332 − 0.0514]
Back	−0.0002 [−0.0013 0.0008]	0.0148 [0.0137 0.0159]	−0.2828 [−0.3716 − 0.1939]

Table 7.2: Parameters of the yield prediction models for the loin and 18cm back products.

implementation on a 64bit PC with 4 dual core AMD 1.8GHz processors, and 32GB RAM.

7.7.2 Virtual Cuts

After registration of each of the 57 carcasses the virtual cuts are performed. Each carcass is cut into three parts; a front end, a middle end and the legs. The middle is then cut into a loin and belly product and a 18cm back and belly product. The cuts are made as described in section 7.5. A qualitative estimation of the cuts are made by visual inspection of the results. Examples of the virtual cuts can be seen in section 7.9.1

7.7.3 Yield Prediction Models

The main goal of this study is to predict the yield of the loin and the 18cm back product using virtual cuts. Yield is defined as the fraction of meat in the product. This is calculated as the weight of the meat divided by the weight of the product. The weight of the bones in the product is not included. A model as described in equation 7.13 is fitted to the calculated yield of the loin and back. The results are seen in table 7.2. The total carcass weight is found to be not significant and is subsequently left out. Histograms of the weight and lean meat percentages used as input are depicted in figures 7.2 and 7.3. Table 7.3 shows the parameters for an reduced model containing only the lean meat percentage as predictor. The model has an explained variance of $R^2 = 0.90$ and $R^2 = 0.93$ of loin and 18cm back yield respectively. Figure 7.18 shows the fit of the models to the data.

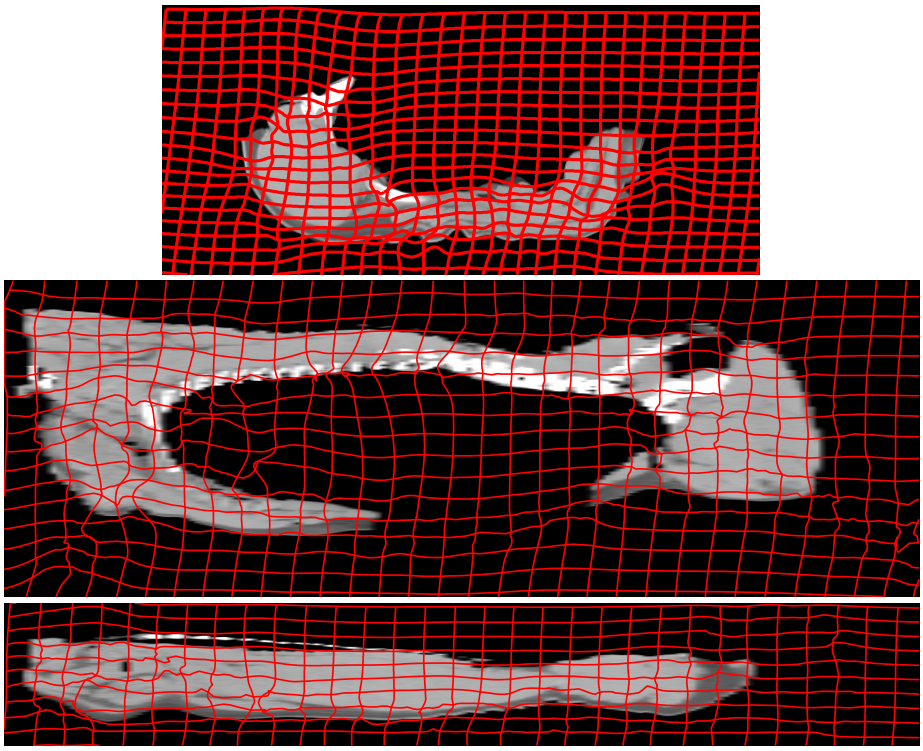


Figure 7.17: The atlas pig with deformation fields. Top row depicts the midaxial plane, middle row the midsagittal plane, and the bottom the coronal plane.

	β_2	β_3	R^2
Loin	0.0130 [0.0119 0.0141]	−0.1169 [−0.1924 − 0.0413]	0.90
Back	0.0148 [0.0137 0.0159]	−0.2929 [−0.3661 − 0.2196]	0.93

Table 7.3: Parameters of the yield prediction models for the loin and 18cm back products using only the lean meat as predictor.

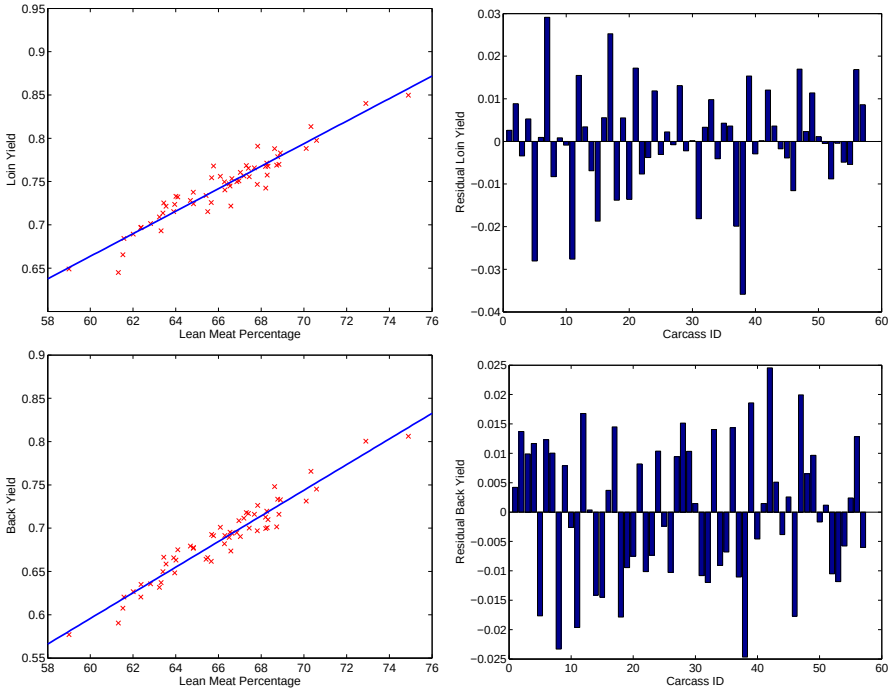


Figure 7.18: The left plot depicts yield of the loin(top) and the 18cm back vs. the lean meat percentage. The right plot depicts the residuals when predicting the yield.

7.8 Discussion and Conclusion

This paper present a preliminary study in the use of CT as a tool for the prediction of yield in cuts of a pig carcass. The errors of the virtual cuts made on the carcasses depend on a number of factors. One important cause of error is of course the registration accuracy. This is measured using the landmarks with a mean Euclidean distance in the training set. This best result is reported to be $\mathcal{D}^{lm} = 6.3$mm which is acceptable bearing in mind that the voxel dimension is $1 \times 1 \times 10$. Furthermore considering the voxel dimensions the placing of the landmarks is very error prone. A measure of the uncertainty of the landmarks is not obtained but by using multiple experts to annotate and following [11, 95] this is clearly possible and should be done in further studies. The effect of misalignment due to registration inaccuracy and landmark errors on the yield estimate should also be investigated. Other more elaborate methods of landmark locating schemes, such as feature detection methods or segmentation based methods, could also have been chosen. However, the choice of image registration has two distinct advantages. One is the ability to process a whole batch of carcasses once the registration has been performed, and the other is that the set of landmarks or segmentation can be altered in the atlas without the need for a new set of registrations. This enables the implementation of other cuts rather quickly.

In this preliminary study the cuts defined are not real-life cuts but rather an adaptation. In further studies the actual definitions of the various cuts must be implemented, however such knowledge is considered proprietary by the slaughterhouses and not suitable for publication.

The prediction models based solely on the lean meat percentage show acceptable results. However a comparison with actual cuts made by a trained butcher or robot is needed in order to access the validity of the results.

All in all the study shows potential for the use of CT and image registration as means for the automatic non-invasive prediction of yield in commercial cuts of pig carcasses. It is possible to calibrate the sorting models presently used in the slaughterhouses with a much larger quantity of carcasses than it is today. But also the search of other predictors such as fat layer thickness etc. is much more feasible using this method. However, ultimately the success of this method depends on how accurately real-life cuts can be modeled.

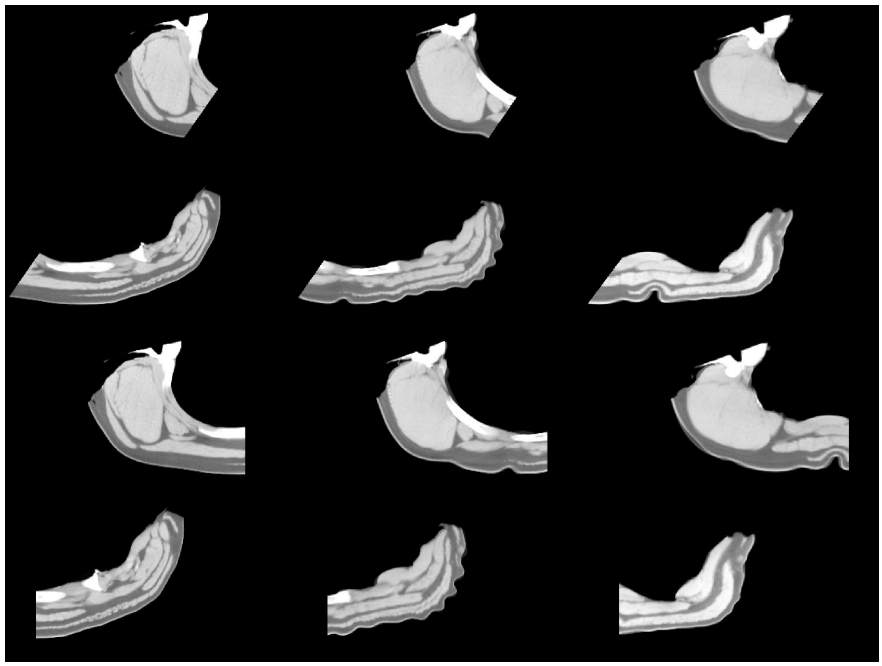


Figure 7.19: The top two rows depicts three slices of the loin and belly. The bottom row depicts three slices of the 18cm back and belly.

7.9 Appendix

7.9.1 Images of Virtual Cuts

This section displays images of the virtual cuts from the test set. They are examples of images used in the qualitative evaluation of the cuts.

Figures 7.19 to 7.22 depicts the cuts of the loin and 18cm back products in four carcasses. The first row is the loin and the second row the corresponding belly product. Third row is the 18cm back and fourth the corresponding belly. Three slices is shown. The images in the first column is from the middle of the anterior part of the product, the second column is the middle slice and the third is from the posterior part of the product.

Figure 7.23 depicts the same carcasses split into the three main parts.

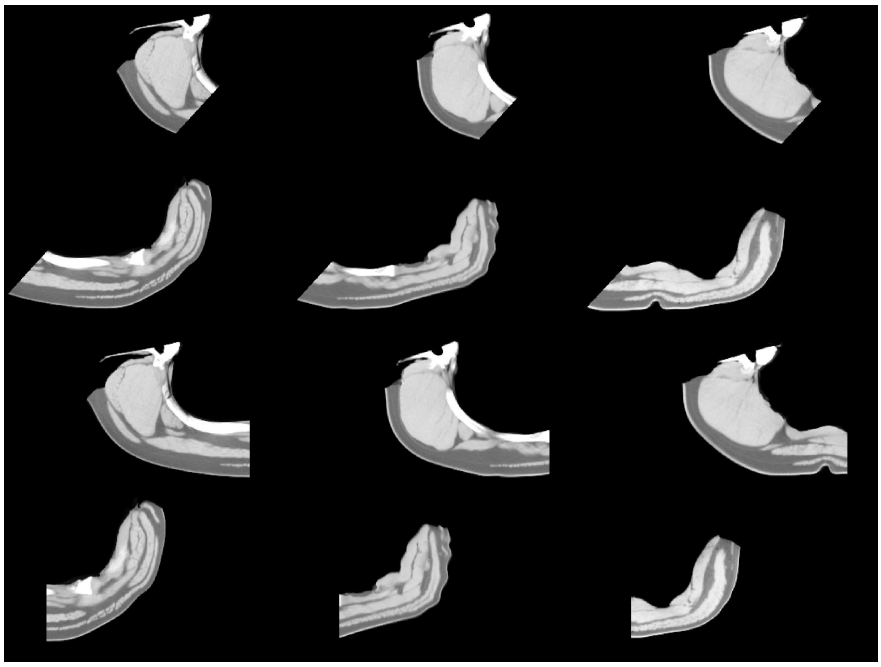


Figure 7.20: The top two rows depicts three slices of the loin and belly. The bottom row depicts three slices of the 18cm back and belly.

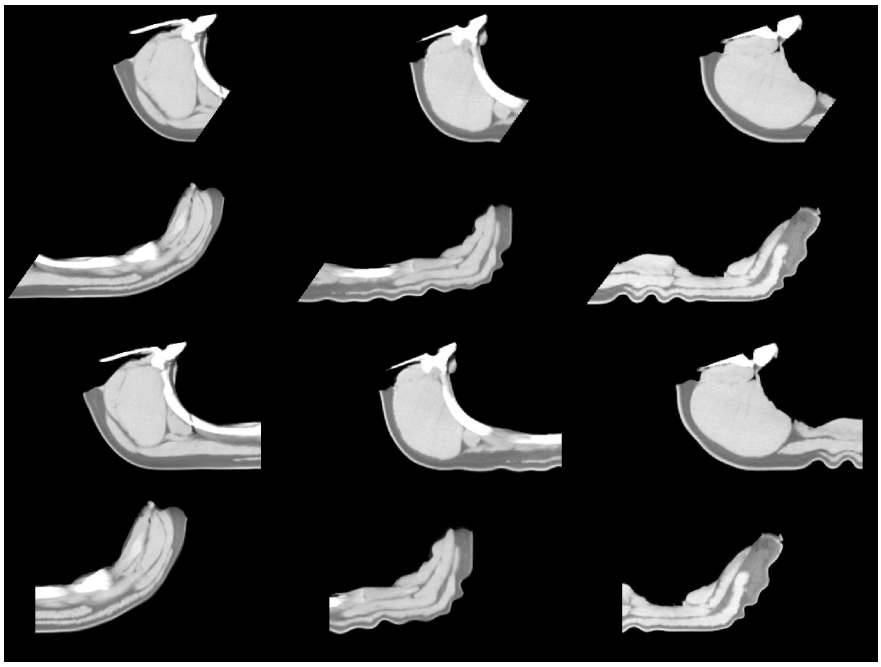


Figure 7.21: The top two rows depicts three slices of the loin and belly. The bottom row depicts three slices of the 18cm back and belly.

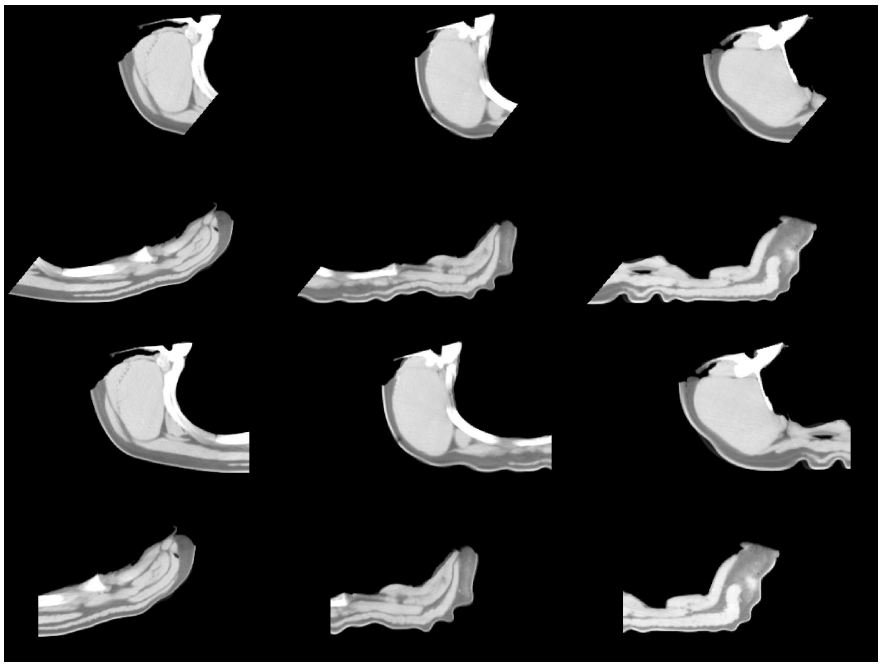


Figure 7.22: The top two rows depicts three slices of the loin and belly. The bottom row depicts three slices of the 18cm back and belly.

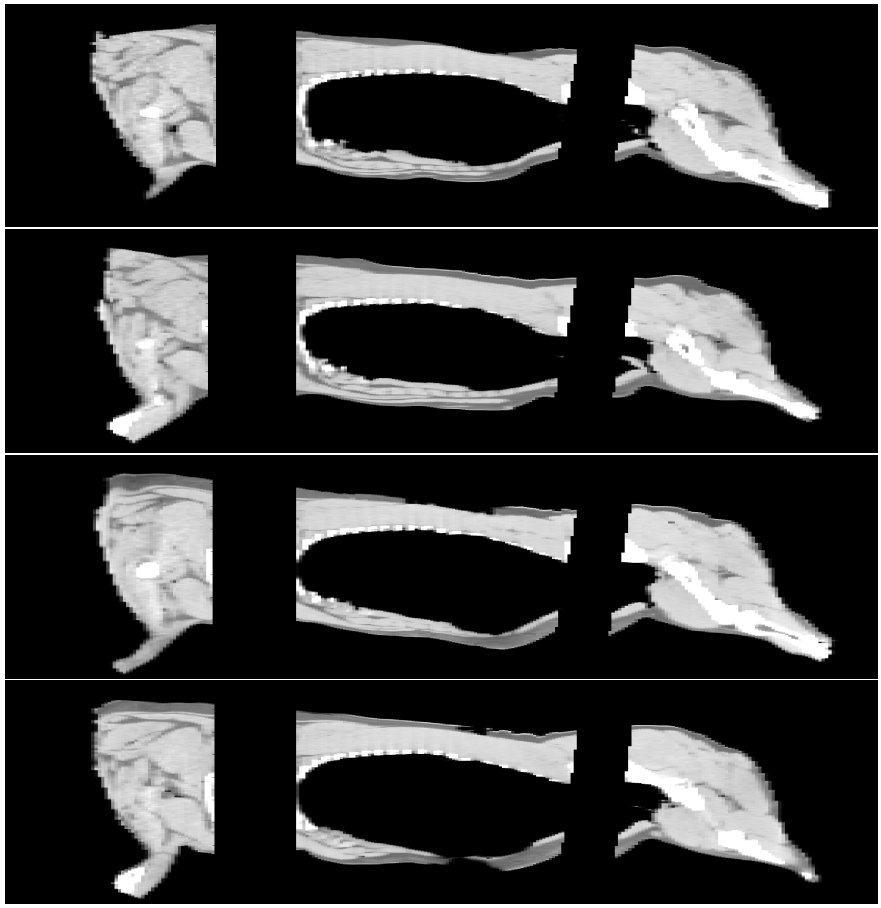


Figure 7.23: Three carcasses split into three parts.

Part IV

Contributions

Chapter 8

Robustness of weight and meat content in pigs determined by CT

L.B. Christensen, M. Vester-Christensen, C. Borggaard and E.V. Olsen

Abstract

An increasing number of European countries are exploring the application of computer tomography (CT) as objective reference technology for determination of lean meat percentage (LMP) in domestic animals. One important requirement for a reference is the reproducibility or quantitative performance. 23 carcasses are CT scanned twice using different settings. The first setting was equal for all carcasses (140 kV, 140 mA, helical, standard reconstruction) whereas for the second scanning either photon current, reconstruction algorithm, or physical position of carcass was changed. The weight determination was based on a volumetric method using specific constants for lean meat, fat and bone tissue thus leaving any difference to be a volume issue. The lean meat content was a simple calculation based on the tissue weights. Our results on weight determination showed that the soft reconstruction algorithm overestimated the weight of the carcass by 0.3% compared to the standard reconstruction. The reduction of photon current down to 80 mA or repositioning of the carcass showed no significant influence on the weight. When calculating LMP no of the introduced changes

in CT protocol introduced significant effect on the estimate of LMP, thus leaving CT as a very robust reference method for determination of LMP in pigs.

8.1 Introduction

Estimation of the lean meat percentage (LMP) is a central part of determination the value of a pig carcass. It is important to the farmer as he is paid by carcass weight and LMP, it is important to the abattoir as the final production yield is influenced by the LMP and an important sorting procedure based on LMP is often employed to optimize the yield. As carcass payment is under EU-regulation objectivity and transparency of the reference for the estimation process is of major importance. Today the reference is based on a manually dissection of commercially important parts of the carcass and thereby the reference is influenced by operator skills and cutting tradition [38]. One way of coping with this problem is to apply a virtual dissection to the carcasses using a full-body CT scanning and a software tool to estimate the LMP_{CT} of the individual carcass. Two different strategies to determine the LMP_{CT} has been proposed: A spectral calibration where the distribution of all voxel densities measured on the Hounsfield scale is calibrated to the result of a manual dissection process using multivariate models. This procedure includes the cutting tradition in the calibration [38]. One other strategy is based on contextual volume grading of all voxels into three different tissue classes: fat, meat or bone [84] The latter has the important advantage of not being influenced by cut-ting tradition nor operator skills. Therefore we apply this strategy.

8.2 Material and Methods

23 pig carcasses was CT scanned twice using two different settings of the CT scanner (GE HiSpeed CT/i) The carcasses were 24 hours PM, prepared for a manual dissection according to EU recommendation, i.e. cutting the head and the feet and removing eventually internal fat and spine cord remains. The scanning was performed twice in a chilled laboratory temporarily attached to the cooling room of the abattoir. One first scanning protocol was used for all carcasses and one second protocol was changed with respect to one of three different scanning parameters according to table 1. The constant protocol parameters are: 140kV volt-age, $0.9 \times 0.9 \times 10mm$ voxel size, 0.7 mm spot size and 10 mm between slice centers. The scanned tomograms were analyzed with a contextual based Markov Random Fields type of algorithm called Owen-Hjort-Mohn [84]. The algorithm estimates the volume of lean meat, fat and bone

Parameter	First	Second					
Protocol No.	1	2	2	2	2	2	2
Reconstruction	Std	Soft	Detail	Std	Std	Std	Std
X-ray current[mA]	140	140	140	140	140	80	100
Position	1	1	1	2	1	1	1
Axial/Helix[A/H]	H	H	H	H	A	H	H

Table 8.1: An overview of the applied scanning protocols. First protocol (No. 1) is arbitrarily used as reference.

tissue in the scanned volumes, i.e. the half carcasses. The three tissue volumes are then multiplied with specific tissue densities to give an estimated carcass weight (WCT). The estimated weight is compared to a scale weight (WS) measured just before scanning and the three tissue densities are found by a linear procedure based on posterior probabilities for each tissue type given a specific voxel [84]. From the tissue weights the LMP_{CT} is estimated for all scanning protocols and the difference is tested for significance together with the difference in estimated carcass weight (WCT).

The two different scanning sessions are made without changing the position of the carcass, the procedure is controlled completely from an adjacent control room outside the scanning area. However, the scanning in 2. position (No. 4) is made after taking the carcass to the chilling room of the abattoir and back to the scanner laboratory.

8.3 Results and Discussions

The first scanning protocol (No. 1) used equally for all carcasses is used as reference and the second (No. $2-7$) is then tested for difference with respect to estimated carcass weight W_{CT} and LMP_{CT}.

8.3.1 Weight differences

Recently [24] the potential of weight estimation has been revealed based on a small test sample and spectral calibration. In the present study we have replaced the spectral calibration with a volume grading and the specific density estimation. We have tested the sensitivity to more scanning parameters ac-cording to table 8.1 above. In figure 8.1 a plot of the estimated weights, calculated with scanning protocols No. $2-7$, are shown with the first protocol (No. 1) as reference.

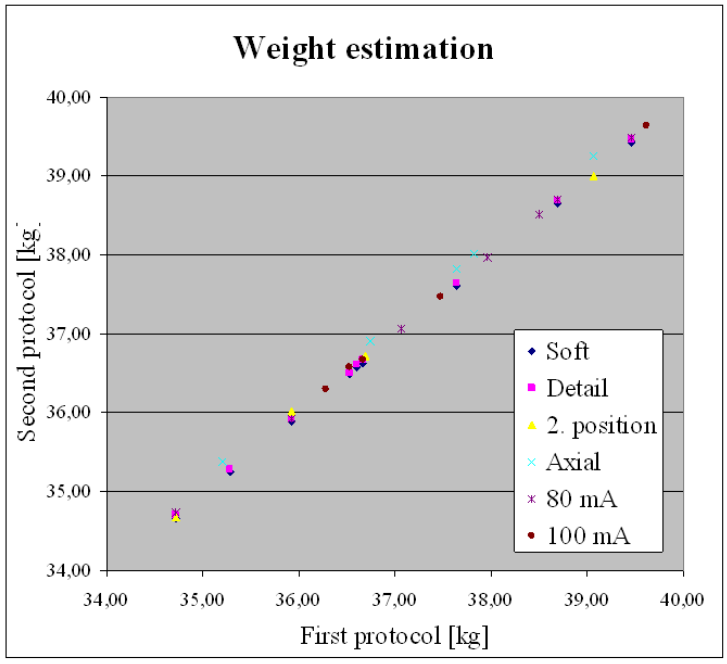


Figure 8.1: Estimated carcass weight using different scanning protocols. Protocol 1 is used as reference.

Protocol difference	Soft	Detail	2.position	Axial	80 mA	100 mA
Protocol No.	2	3	4	5	6	7
Difference [kg]	0.041	0.007	0.001	−0.18	−0.009	−0.017
Difference [%]	0.11	0.02	0.00	−0.48	−0.03	−0.05
Significance p	0.0000	0.0570	0.9761	0.0000	0.0209	0.0955
Significance level	***	NS	NS	***	*	NS

Table 8.2: Weight estimation.

Protocol difference	Soft	Detail	2.position	Axial	80 mA	100 mA
Protocol No.	2	3	4	5	6	7
Difference [LMP_{CT} %]	−0.02	0.03	−0.10	0.17	−0.05	0.04
Significance p	0.3108	0.1285	0.2620	0.0074	0.1990	0.2943
Significance level	NS	NS	NS	**	NS	NS

Table 8.3: LMP_{CT} estimation.

The average difference between weight estimates from the first protocol to the second protocol is calculated and tested using a standard t-test. The results show a significant difference in weight estimate of less than 0.5% irrespective of which of the seven different scanning protocols that is applied to the scanning procedure. This is of the same level as estimated for the used scale.

8.3.2 LMP_{CT} estimation

The sensitivity of the LMP_{CT} estimation to the scanning protocol is evaluated and the results shown in figure 8.2. The correlation seems even better than the performance for the weight estimation. Again the difference between the LMP_{CT} estimated with the different scanning protocols of table 1 is evaluated and tested for significant difference using a standard t-test. The results listed in table 8.3 show that LMP_{CT} estimation only has a slight dependence ($**$) of the changes in scanning protocol applied in this experiment. The most significant difference is the change of Helix to Axial scanning where a difference ($p = 0.0074$) is found.

8.4 Conclusion

From this experiment it may be deduced that CT is capable of measuring carcass weight and estimate the lean meat content in a quite robust and objective way. The highest sensitivity of scanning protocol is found in weight estimation where a relative error of less than 0.5% of changing a Helix scanning to an Axial

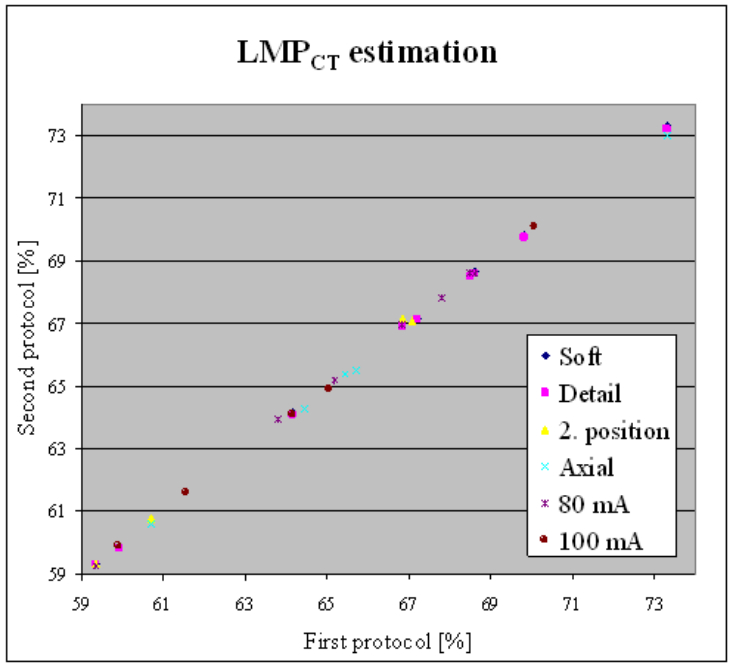


Figure 8.2: Estimated LMP_{CT} using different scanning protocols. Protocol 1 is used as reference.

scanning protocol. Minor sensitivity (but still significant) is found by change of reconstruction algorithm from a standard to a soft and no significance influence in repositioning the carcass, reduction of photon current from 140 mA to 100 mA or reconstruction with a high contrast detailed algorithm. Determination of the lean meat content, expressed as LMP_{CT}, is demonstrated to be even less sensitive to the imposed changes in scanning protocol. For estimation of this very important parameter the experiment reveals that CT may be expected to have only a slight dependence to changing the scanning from Helix to Axial. The remaining changes in reconstruction, positioning and applied x-ray current result in not significant changes in the LMP_{CT} determination. This concludes that CT offers a robust measurement technology with a very small sensitivity to the different settings of the scanning protocol.

8.5 Acknowledgements

We thank our colleagues Jesper Blom-Hansen, Mads Fogtmann Hansen, Søren Hemmingsen Erbou and Peter Vorup for their dedication and attention to detail in this experiment.

Chapter 9

Virtual Dissection of Pig Carcasses

Martin Vester-Christensen, Søren G. H. Erbou, Mads F. Hansen, Eli V. Olsen, Lars B. Christensen, Marchen Hviid, Bjarne K. Ersbøll and Rasmus Larsen

Abstract

This paper proposes the use of computed tomography (CT) as a reference method for estimating the lean meat percentage (LMP) of pig carcasses. The current reference is manual dissection which has a limited accuracy due to variability between butchers. A contextual Bayesian classification scheme is applied to classify volume elements of full body CT-scans of pig carcasses into three tissue types. A linear model describes the relation between voxels and the full weight of the half carcass, which can be determined more accurately than that of the lean meat content. 299 half pig carcasses were weighed and CT-scanned. The explained variance of the model was $R^2 = 0.9994$ with a root mean squared error of prediction of 83.6g. Applying this method as a reference will ensure a more robust calibration of sensors for measuring the LMP, which is less prone to variation induced by manual intervention.

9.1 Introduction

Throughout the European Union (EU) the lean meat percentage (LMP) is used for classifying pig carcasses and is defined as the ratio of weighed lean meat versus the weight of the pig carcass. Measuring the LMP is typically done using ultrasound or optical sensors which are calibrated towards a common manual dissection method of half pig carcasses, cf. Commission of the European Communities (EC) [27] and Walstra and Merkus [134]. The accuracy and precision

of these calibrations are limited by that of the dissection method. Only highly trained butchers are involved in such a dissection. Even so there is still a significant difference between butchers as reported by Nissen et al. [92]. The maximum difference in estimated LMP between 8 butchers is found to be 1.96 LMP units and the jointing of the carcasses is found to be a critical point in the EU dissection method. Furthermore variation between countries were also found. Olsen et al. [97] report that in general variations between butchers is more important than variations between copies of the same type of instrument, when calibrating instruments to manual dissection.

X-ray computed tomography (CT), cf. Cho et al. [21], is a non-invasive technique that measures the radio-density of a material, i.e. the relative attenuation of X-rays through the material and is measured in the Hounsfield scale. The scale is calibrated such that air is at -1000 Hounsfield Units (HU) and water at 0HU, making HU-values comparable across scanners and settings. Fat tissue is usually around -60HU, meat tissue around $+60$HU and bone tissue above ~ 150HU. The CT-volume consists of discrete volume elements (voxels) and are not necessarily isotropic. Voxels might also consist of more than one class of tissue. The latter is denoted partial volume effects (PVE) and results in overlapping probability density functions (pdf) of the different tissues. Figure 9.1 shows a typical histogram in the fat/meat range for a CT scanned pig carcass. The left peak represents fat and the right peak represents meat. Bone is above the range shown.

The fixed Hounsfield scale of CT is a major reason for using CT instead of magnetic resonance imaging (MRI) because it is comparable across scanners. Applying different settings, or protocols, in a specific CT-scanner has been shown by Christensen et al. [23] to give quite robust results w.r.t. LMP. Based on 23 pig carcasses and using 7 different protocols they find a maximum difference of 0.27 LMP units and a maximum difference in the estimated carcass weight of 0.22kg.

Typically a simple threshold in the CT histogram is used to distinguish fat, meat and bone tissue, but this will often result in errors caused by noise in the reconstruction, artifacts and PVE.

Several attempts have been made on calibration of CT scans of pigs carcasses to predict the lean meat content of manual dissections. Glasbey and Robinson [46] derive and compare estimators of tissue volumes in CT images taking mixed pixels, or PVE, of fat and meat into account. A moment-based estimator performs best in both a simulation study and in a particular application where tissue composition of sheep is estimated. The improvement in precision is reported to be minor compared to Cavalieri sampling, cf. Roberts et al. [101].

Dobrowolski et al. [34] and Romvári et al. [106] use thresholds in the histogram

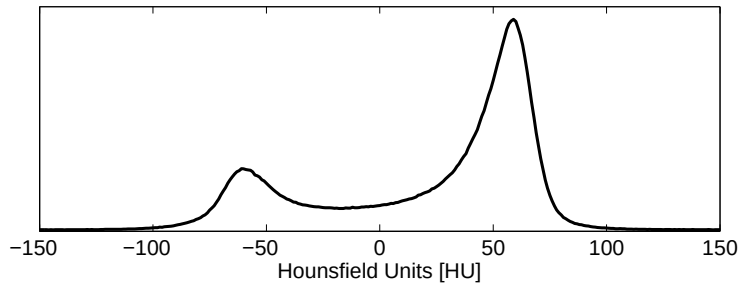


Figure 9.1: Histogram of a CT-volume of a pig carcass. The ordinate is scaled to show the distribution of fat (left) and meat voxels (right).

of CT and Collewet et al. [25] of MRI scans to segment meat voxels. In these studies partial least-squares regression (PLSR) of histogram values is applied to model the dissected lean meat content. Table 9.1 summarizes their results along with those of Johansen et al. [68]. R^2 is the explained variance and RMSEP/C are the root mean squared errors of prediction/calibration. Johansen et al. [68] apply thresholds to the histogram of 15 anatomically chosen slices of 120 CT-scanned carcasses of lamb to segment fat and meat tissue. A multidimensional PLS model is applied on the histogram values of fat and meat to predict the corresponding weights in a manual dissection. The RMSEP of the meat content is reported to be 772g before and 561g after bias correction, with an $R^2 = 0.96$. Common for the above mentioned methods is that they only take into account the histogram value of the voxel to be classified and not any of the neighboring voxels.

Lyckegaard et al. [84] apply a multivariate Bayesian 2D contextual classification scheme to each slice as described by Larsen [78]. Certain combinations of neighboring voxels are taken into account modeled in a Bayesian scheme with priors obtained from thresholds in the histogram. Linear regression is used to estimate the parameters of a model mapping the volume of fat, meat and bone to the total weight of the carcass, with an $R^2 = 0.991$ and a RMSEP= 584g.

This paper presents an experiment consisting of 299 pig carcasses, which are weighed and CT-scanned. Applying methods from image processing along with a contextual classification scheme the CT-volume is classified into several types of tissue. A linear model determines the mapping from voxels to the full weight of the half carcass, which is then used for estimating the CT-based LMP.

Paper	[34]	[25]	[106]
Modality	CT (full, 150 sl.)	MRI (full)	CT (full)
Vox/spac. [mm]	−/−	$[0.77, 1.02, 8]/10$	$[\sim 1, \sim 1, 10]/10$
Comment	1/2 pig carc.	1/2 pig carc.	1/2 pig carc.
Amount	60	120	60
R^2	0.990	−	0.992
RMSEP/C [g]	270/−	465/400	−/232
Bias [g]	16	−	−

Paper	[68]	[84]	
Modality	CT (15 anat. sl.)	CT (full)	
Vox/spac. [mm]	$[0.78, 0.78, 3]$/var.	$[1, 1, 10]/10$	
Comment	Lamb carc.	1/2 pig carc.	
Amount	120	57	
R^2	0.961	0.991	
RMSEP/C [g]	772/−	584/554	
Bias [g]	530	−	

Table 9.1: Previous work. Papers [34], [25], [106] and [68] apply PLS-methods on histograms for meat pixels, modeling the lean meat weight obtained from dissection. [84] apply a contextual Bayesian classifier and linear regression for predicting the full weight of half carcasses. R^2 is the explained variance, RMSEP/C are the rms errors of prediction/calibration, with the corresponding bias reported in some cases.

9.2 Materials and Methods

9.2.1 Data

299 carcasses representing the Danish pig population with respect to weight (warm slaughter weight) and fatness (fat depth between the 2nd and 3rd hindmost thoratic vertebra) were selected. Half of which were gilts and the rest castrates. The pigs were slaughtered at a commercial Danish abattoir and cooled. The day after slaughtering the left side of the carcasses were prepared for dissection. The preparation was done according to Walstra and Merkus [134], but the head except the cheek and toes were cut off before scanning. All half carcasses were weighed on a DIGI DS160 industrial scale with an accuracy of 20g. Subsequently they were scanned with a GE HiSpeed CT/i single-slice scanner. In the following the term carcass weight denotes the weight of the scanned left side of the carcass. The scanning protocol parameters were: 140kV voltage, $0.9 \times 0.9 \times 10$mm voxel size, 0.7mm spot size and 10mm between slice centers, yielding 299 CT-volumes of pig carcasses with corresponding weight. Figure 9.2 shows a left side of a carcass prepared and ready for scanning.

Figure 9.2: Left side of a carcass prepared and ready for scanning.

9.2.2 Full Dissection

Of the 299 carcasses scanned, a subsample of 29 carcasses with 13 gilts and 16 castrates were selected. The subsample was selected representing the distribution of weight and fatness. After scanning a full dissection was made on the same carcass to calculate the lean meat content. The LMP is defined as the ratio of the meat and the total weight of the carcass exclusive head and toes. Full dissection is not standardized yet. In this trial the meat fraction consists of all muscles including tendons, fascia and periosts. Periosts appear by e.g. extraction of ribs, femur bone in ham and front part. Tendons from certain muscles stretch around the bones as e.g. *Biceps brachii* and other muscles in the front part and ham. These tendons are not left entirely on the muscles, but are cut off where they touch the bone. The fat fraction consists of subcutaneous and inter-muscular fat including skin and glands, veins and loose membrane tissue. Loose membrane tissue is defined as all membrane tissue which can be lifted between two fingers and can be cut without damaging the underlying muscle. The bone fraction consists of all bones including cartilage. No bones are scraped to remove periosts or remains of tendon.

9.2.3 Tissue Classification

For identifying meat voxels, the tissue from CT is traditionally classified by applying thresholds in the histogram. This method introduces errors due to PVE as mentioned earlier. In the current work a multivariate Bayesian 2D contextual classification scheme is applied to each slice, cf. Larsen [78]. Background voxels are removed and tissue voxels are classified into three classes; fat, meat and bone. The classifier takes certain configurations of neighboring voxels into account as well as the prior probability as described in Lyckegaard et al. [84]. All fat,

meat and bone tissue irrespective of their anatomical position are regarded as belonging to the same corresponding class. As a postprocessing step the bones are morphologically closed such that marrow will be part of the bone class. In CT skin voxels are more similar to meat. When comparing the LMP obtained by CT to that obtained by manual dissection the skin is segmented separately and considered as fat such that the LMP can be computed according to Commission of the European Communities (EC) [27]. Segmentation of the skin is done using mathematical morphology, cf. Gonzalez and Woods [48].

9.2.4 Density Estimation

Estimating the weight of a carcass requires an approximation of the densities ρ of the tissue types in every voxel. The carcass weight is modeled as a linear combination of the weights of the tissue classes. Labeling of a particular voxel is done by choosing the class with *maximum a posteriori* (MAP) probability, see Larsen [78]. The MAP model applied for a single carcass with three tissue classes is,

$$w_i = \rho_f n_f v + \rho_m n_m v + \rho_b n_b v + \epsilon_i, \tag{9.1}$$

where v is the voxel volume, n_f, n_m and n_b are the number of voxels classified as fat, meat and bone respectively. w_i is the measured i^{th} carcass weight and $\epsilon_i \in N(0, \sigma_i)$. Including all carcasses and using linear regression the density approximations can be obtained.

Due to PVE a single voxel might consist of more than one type of tissue. However, in the model in eq. (9.1) each voxel is labeled as either fat, meat or bone. Including PVE in the model can be done using the value of the posterior probability of each class. Thus all voxels have a weighted contribution from all classes.

Figure 9.3 illustrates the issues with PVE. The figure depicts a slice in the shoulder part of the carcass where voxels with a posterior probability above 0.5 and below 1 of belonging to the meat class are yellow, indicating that they contain something else than meat. These are primarily located where the meat interfaces with fat. Integrating PVE in the carcass weight model yields,

$$\begin{aligned} w_i &= \rho_f \textstyle\sum_{i=1}^{n} p(c_f|x_i)v + \rho_m \sum_{i=1}^{n} p(c_m|x_i)v \\ &+ \rho_b \textstyle\sum_{i=1}^{n} p(c_b|x_i)v + \epsilon_i, \end{aligned} \tag{9.2}$$

where n is the total number of voxels. $p(c_f|x_i)$, $p(c_m|x_i)$ and $p(c_b|x_i)$ are the posterior probabilities of voxel x_i belonging to the fat, meat or bone class respectively, and $\epsilon_i \in N(0, \sigma_i)$. Both the MAP and the PVE model are applied with and without an additional constant term c, for comparison.

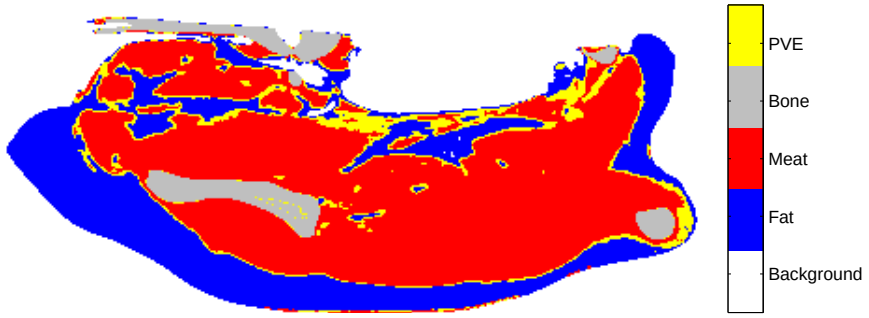


Figure 9.3: Partial volume effects shown in a CT-slice from the shoulder part of half a pig carcass. Yellow denotes voxels with a probability above 0.5 and below 1.0 of belonging to the meat class.

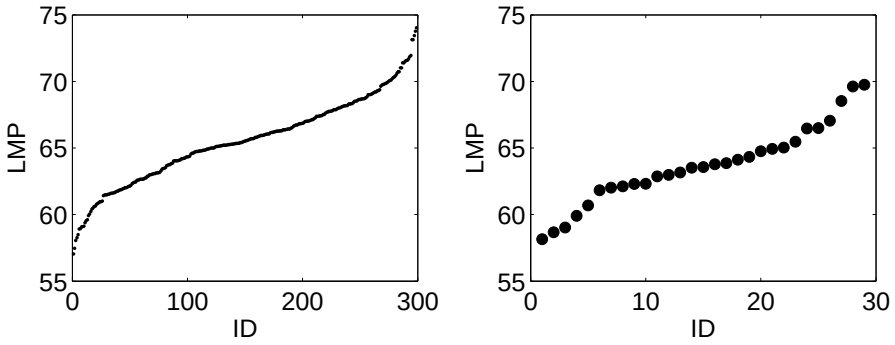


Figure 9.4: The resulting LMP estimated by CT, 299 carcasses (left), and by manual dissection, 29 carcasses (right), sorted by LMP.

To avoid the effect of outliers the linear regression problem is solved using an iteratively re-weighted least squares algorithm presented in Holland and Welsch [64]. Leave-one-out cross-validation is performed and the root mean squared error of the residuals of prediction (RMSEP) is reported as well as the bias and explained variance (R^2).

Tissue Type	Fat	Meat	Bone
Res. mean±std [%]	2.49 ±0.55	−3.07 ±0.57	0.58 ±0.33
Res. mean±std [g]	968 ±181	−1227 ±210	227 ±130

Table 9.2: Mean and standard deviations of the residuals obtained by comparing CT dissection with manual dissection.

9.3 Results and Discussion

9.3.1 Comparison with manual dissection

Figure 9.4 shows the range of LMP for both CT (left) and manual dissection (right) and is approximately $[55, 75]$ units. The half carcass weight range is seen in figure 9.6 to be approximately $[31, 49]$kg. Data used in both dissection methods cover the variation in LMP of the Danish pig population. Table 9.2 and figure 9.5 compare the estimated tissue content from the manually dissected carcasses with the corresponding estimate from the CT dissection. On average CT scanning identifies 1227g more meat, 968g less fat and 225g less bone in a carcass than manual dissection. It is expected that tissues like tendons, fascia, periosts and cartilage, which consist of protein, will be considered as meat in a CT scan. From the description of the three main groups of tissue, meat, fat and bone obtained with manual dissection, it is seen that only a part of all protein-containing tissues is defined as meat. It seems reasonable that the limitations of manual separation together with the definition of meat cause the main contribution to the differences between LMP determined with CT and manual dissection. Furthermore table 9.2 indicates a larger standard deviation when compared to the mean value of the residuals of the bone class than for the meat and fat classes.

9.3.2 Modeling total weight

Applying both models described in section 9.2.4 reveal similar results. Figure 9.6 shows a plot of the correlation between estimated carcass weight and measured carcass weight using the MAP model, cf. eq. (9.1). The estimated parameters and correlation results for the MAP model and the PVE model, with and without constant terms c, are reported in table 9.3. In all regressions the robust algorithm detects 5 outliers, which are identified as errors in the data acquisition. These are subsequently removed in the calculation of the parameters and the correlation results as well.

Table 9.3 shows that the four models perform equally well with large correlations

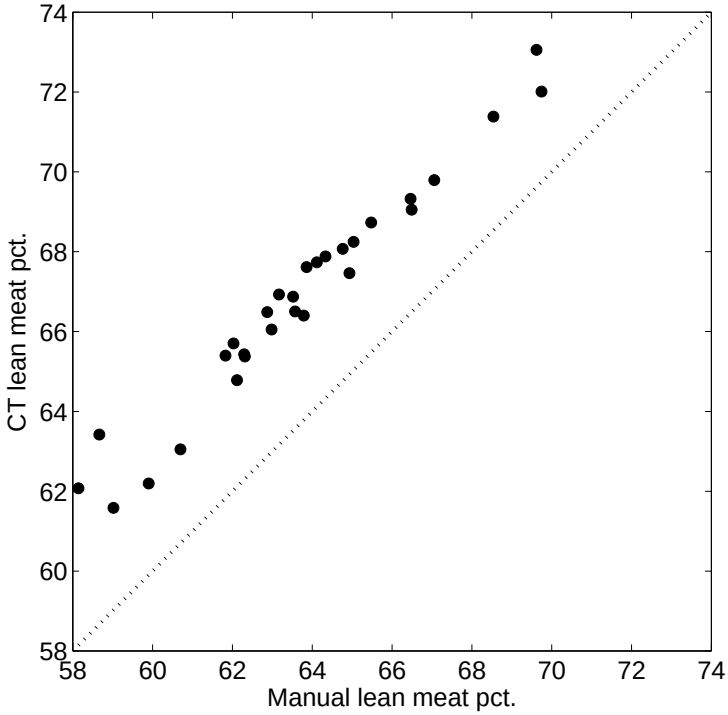


Figure 9.5: LMP estimated by manual dissection versus CT estimated LMP using the MAP model.

Model	R^2	RMSEP [g]	Bias [g]
MAP	0.9994	83.6	2.6
PVE	0.9994	79.0	2.3
MAP+c	0.9994	79.1	1.8
PVE+c	0.9994	75.5	1.7

Table 9.3: Predictive performance of the two models, with and without a constant term c, using leave-one-out cross-validation.

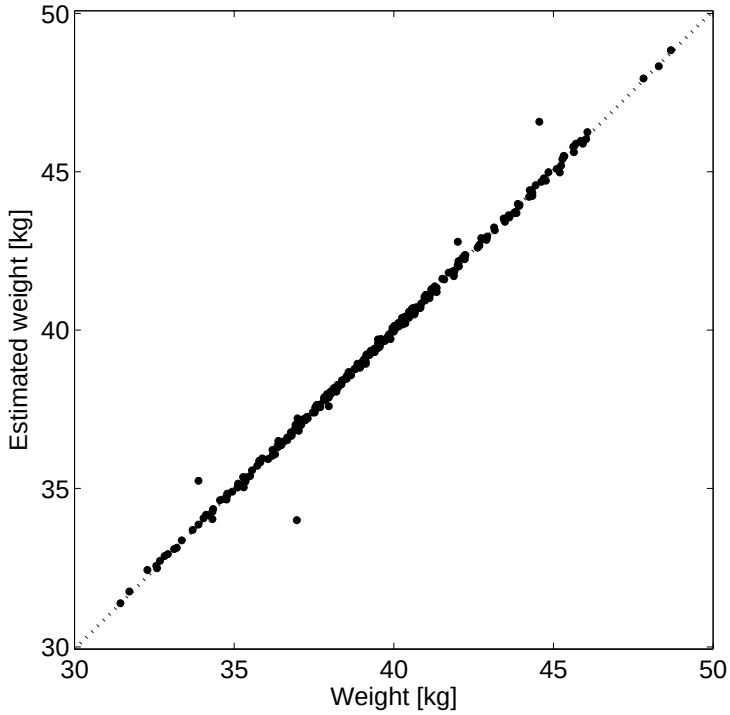


Figure 9.6: Estimated weight using the MAP model versus measured weight.

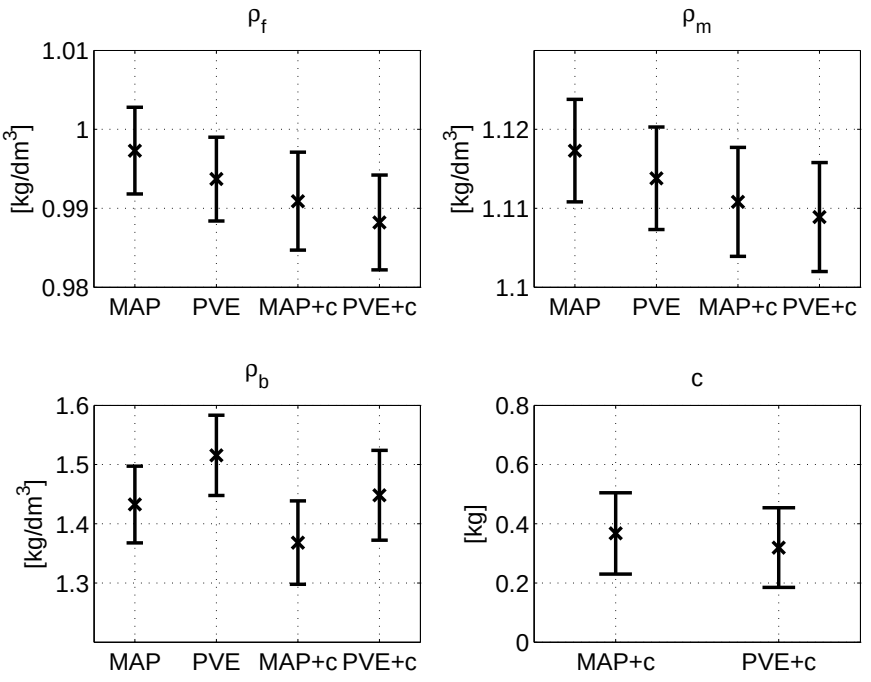


Figure 9.7: Estimated parameters and their corresponding 95% confidence intervals for the two models, with and without a constant term c.

Model	ρ_f [CI]	ρ_m [CI]	ρ_b [CI]	c [CI]
MAP	0.997 [0.992 1.003]	1.117 [1.111 0.124]	1.433 [1.368 1.497]	
PVE	0.994 [0.988 0.999]	1.114 [1.107 1.120]	1.516 [1.448 1.583]	
MAP+c	0.991 [0.985 0.997]	1.111 [1.104 1.118]	1.368 [1.298 1.438]	0.367 [0.230 0.505]
PVE+c	0.988 [0.982 0.994]	1.109 [1.102 1.116]	1.448 [1.372 1.524]	0.319 [0.185 0.454]

Table 9.4: The resulting parameters for the MAP and PVE models excluding and including a constant term c. 95% confidence intervals are shown in brackets.

to the measured weight. Applying a one way analysis of variance (ANOVA) on the weight estimates from all models reveals no significant difference between them. Including a constant term would make the definition of the LMP ambiguous, since it does not belong to a specific tissue class. Subsequently the simple MAP model without a constant term is preferable. Modeling PVE has no effect on the quality of the predicted weight. In a randomly chosen carcass only 1.6% of all the voxels classified as meat have a fat probability above 0.1. Thus the influence of PVE is very limited with regards to the total weight. Table 9.4 and figure 9.7 show that the values of the parameters of fat and meat are not significantly different when comparing the PVE and MAP models contrary to the bone parameter. A voxel containing both bone and soft tissue will tend to be classified by the MAP model as bone. A voxel in the PVE model contributes to all tissue types. This results in more bone voxels using MAP than using PVE.

All in all the results obtained are very encouraging when compared to table 9.1. The simple MAP based model has an explained variance of $R^2 = 0.9994$, a bias of 2.6g and RMSEP=83.6g estimated using leave-one-out cross-validation.

For all models the three tissue types are assumed to have the same properties regardless of their anatomical position. Thus the parameters ρ_f, ρ_m, and ρ_b can be viewed as the average density of all fat, meat and bone in the half carcass. Previous work (Romvári et al. [106]) reports the importance of modeling different tissue properties, and they do this by manually separating the CT-volume into three carcass parts. This is prone to operator dependent errors. In the present study it is argued that using average tissue properties yields a more robust estimate of the carcass weight due to operator independency. It should be noted though, that the parameters might not have a strict physical interpretation as densities of the specific tissue classes.

Even though there is a clear definition of which of the three tissue fractions the tendons and glands etc. belong to, the specific butcher makes the final decision. Nissen et al. [92] report considerable variation between butchers and separation of muscles and especially small muscles are very dependent on the butcher. The contribution from the butchers affects mainly the precision of dissection and less the average result. Two main sources of error are present when calibrating online instruments to LMP. One is the error or variation, which expresses the

imperfect relation between the reference LMP and the online measurements, including the accuracy of the online measurements, and the other one is the accuracy of the dependent variable i.e. the reference LMP.

LMP based on CT is a very promising candidate for an instrumental reference for pig carcass classification. Previous investigations have shown very high repeatability. However, before CT LMP can be used as a global reference, it has to be documented that the results can be reproduced independently of CT instruments, time and pig population. The method described in this paper is based on a specific scanning protocol and reconstruction algorithm. Although the method seems robust to these factors a thorough documentation will be necessary. Especially the choice of slice thickness, resolution and reconstruction algorithm has to be general and available on all types and makes of CT scanners. A possible tool to ensure the reproducibility over time, including a possible bias correction, could be calibration using phantoms that mimic different types of carcasses with known values of LMP. How such phantoms should be designed is an area of future research.

Replacing the manually determined LMP with CT based LMP will improve the calibration problem significantly, even though the lack of a perfect relationship is an important issue. Disregarding the fixed costs related to the purchase of a CT scanner and installing it in a trailer, the lower costs using CT is a considerable advantage compared to manual dissection. If only the maintenance of the scanner is taken into account alongside the salary of the operators, a CT based LMP costs less than half that of a manual dissection.

9.4 Conclusions

Previous work shows CT-based methods as robust compared to manual dissection, and as such constitute a suitable reference. This work presents a robust and accurate calibration reference, where variation due to manual intervention is minimized. Given a model of the carcass weight, the LMP can be estimated based on the classification of the volume elements (voxels) in the CT volume. Using this more accurate method as a reference will make the calibration procedures of other LMP sensors much more standardized and accurate.

Contextual models based on segmentation of the carcass into three classes is validated on a large data set of 299 half pig carcasses. Incorporating the influence of partial volume effects is found not to be significantly better than a maximum-a-posteriori model. All models correlate very well with the full weight of the half carcasses, with the simple maximum-a-posteriori based model being the model of choice. The model has an explained variance of $R^2 = 0.9994$, a bias of 2.6g

and a root-mean-squared error of prediction of RMSEP=83.6g. These results are very encouraging compared to previous work, for which reason the method is suggested as a new reference for calibration of sensors used for pig carcass grading.

Acknowledgements

The CT data was provided by the Danish Meat Research Institute as a part of the project "The Virtual Slaughterhouse" funded by the Danish Pig Levy Fund and the Directorate for Food, Fisheries and Agri Business.

Chapter 10

Accelerated 3D Image Registration

Martin Vester-Christensen, Søren G. Erbou, Sune Darkner and Rasmus Larsen

Abstract

Image registration is an important task in most medical imaging applications. Numerous algorithms have been proposed and some are widely used. However, due to the vast amount of data collected by eg. a computed tomography (CT) scanner, most registration algorithms are very slow and memory consuming. This is a huge problem especially in atlas building, where potentially hundreds of registrations are performed. This paper describes an approach for accelerated image registration. A grid-based warp function proposed by Cootes and Twining, parameterized by the displacement of the grid-nodes, is used. Using a coarse-to-fine approach, the composition of small diffeomorphic warps, results in a final diffeomorphic warp. Normally the registration is done using a standard gradient-based optimizer, but to obtain a fast algorithm the optimization is formulated in the inverse compositional framework proposed by Baker and Matthews. By switching the roles of the target and the input volume, the Jacobian and the Hessian can be pre-calculated resulting in a very efficient optimization algorithm. By exploiting the local nature of the grid-based warp, the storage requirements of the Jacobian and the Hessian can be minimized. Furthermore, it is shown that additional constraints on the registration, such as the location of markers, are easily embedded in the optimization. The method is applied on volumes built from CT-scans of pig-carcasses, and results show a two-fold increase in speed using the inverse compositional approach versus the traditional gradient-based method.

10.1 Introduction

Registration of images is an important an actively researched area of medical imaging. It is the task of transforming the geometry of two or more images such that their corresponding regions are aligned. The need may arise from comparison of images from different imaging modalities, from images obtained at different times, from different patients, or from comparison with a patient atlas. Registration is needed in a wide variety of medical applications, eg. for diagnostic purposes, for pre-surgery planning or for treatment estimation. The medical imaging hardware of today produce images of high resolution, and as a consequence a huge amount of data need processing in order to solve the registration problem. The requirements on the processing hardware are very high in terms of storage capability, CPU speed, and maybe most importantly in memory capacity.

A range of different registration algorithms have been produced in the recent years. The algorithm described in this paper, belongs to the class of parameterized methods. Hence the registration can be described by a set of parameters. Existing methods includes Rueckert [108] *et al.* using B-splines on a grid to define the warp-field, Cootes [30] *et al* uses bounded diffeomorphisms, warping pixels inside a unit sphere based on the displacement of the sphere center. An example of the non-parametric approach is found in Christensen [22] *et al.* which solve partial differential equations for fluid motions to align images.

This paper presents the acceleration of an image registration algorithm [31] by Cootes *et al.*. An inverse compositional [7] optimization scheme, proposed by Baker and Matthews, is used. It is a Gauss-Newton approach, but in which the Jacobian and the estimated Hessian can be precomputed. However, to be tractable memory wise, this requires exploitation of the properties of the registration algorithm. An additional benefit of the Gauss-Newton approach is the ease of which addition of soft constraints on the registration can be added. Comparison is made with optimization using the Lucas-Kanade scheme.

The paper consist of the following. First the registration algorithm is described. In 10.2.1 the image registration algorithm is described, followed by outlining of the Lucas-Kanade algorithm, in 10.2.2, and the inverse compositional optimization algorithm, in 10.2.3. Addition of soft constraints is described in 10.2.4. Section 10.3 describes the results obtained by comparing the two optimization schemes.

10.2 Methods

The image registration algorithm utilized in this paper is proposed by Cootes [31] *et al.* which builds on the algorithms presented by Rueckert [109] *et al.* The image registration is performed by composing a series of grid-based diffeomorphic warps which ensures the resulting warp being diffeomorphic.

10.2.1 Grid-Based Diffeomorphisms

A grid-based warp is represented by a grid of nodes[1], see figure 10.1. The transformation $\mathbf{W}$ of a pixel $\mathbf{x} = (x, y, z)$ is found by interpolating the displacement $\mathbf{d}$ of its surrounding grid nodes. In 3D the interpolating scheme is as below,

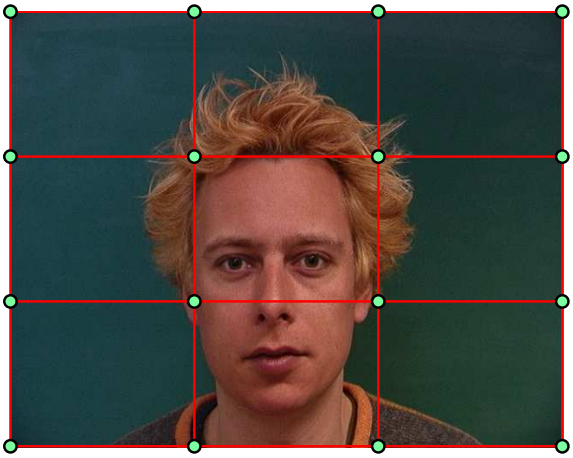
$$\begin{aligned}
\mathbf{W}(\mathbf{x};\alpha) &= k(x-i)k(y-j)k(z-l)\mathbf{d}_{i,j,l} \\
&+ k(i+1-x)k(y-j)k(z-l)\mathbf{d}_{i+1,j,l} \\
&+ k(x-i)k(j+1-y)k(z-l)\mathbf{d}_{i,j+1,l} \\
&+ k(i+1-x)k(j+1-y)k(z-l)\mathbf{d}_{i+1,j+1,l} \\
&+ k(x-i)k(y-j)k(l+1-z)\mathbf{d}_{i,j,l+1} \\
&+ k(i+1-x)k(y-j)k(l+1-z)\mathbf{d}_{i+1,j,l+1} \\
&+ k(x-i)k(j+1-y)k(l+1-z)\mathbf{d}_{i,j+1,l+1} \\
&+ k(i+1-x)k(j+1-y)k(l+1-z)\mathbf{d}_{i+1,j+1,l+1},
\end{aligned} \tag{10.1}$$

where $k()$ denotes a suitable kernel function which is non-zero only for $i \leq x < i+1$, $j \leq y < j+1$ and $l \leq z < l+1$. The warp is parameterized with the components of the displacement vectors $\mathbf{d}$. Thus, with a $3 \times 3 \times 3$ grid in 3D, the warp consist of 81 parameters. The kernel is chosen as $k(r) = \frac{1}{2}(1 + \cos(\pi r))$ which gives a smooth and invertible mapping [31] given that $-\frac{1}{\pi} < r < \frac{1}{\pi}$. Using the interpolating scheme the warp is regularized by the coarseness of the grid. Thus, a pixel cannot move outside the bounding box provided by the surrounding grid nodes. However, to represent a complex transformation several simple warps can be composed,

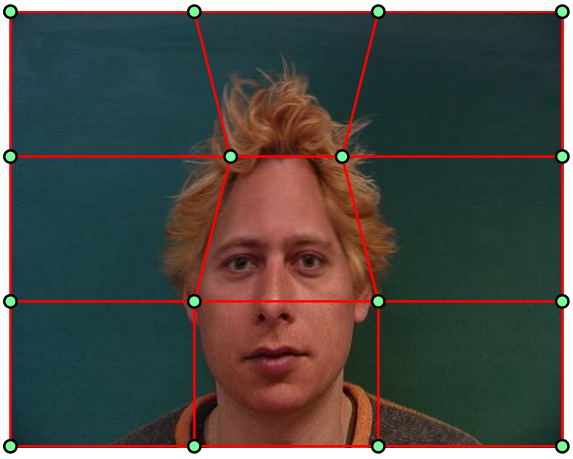
$$\mathbf{W}(\mathbf{x};\mathbf{p}) = \mathbf{W}(\mathbf{x};\delta_1) \circ \mathbf{W}(\mathbf{x};\delta_2) \ldots \mathbf{W}(\mathbf{x};\delta_{n-1}) \circ \mathbf{W}(\mathbf{x};\delta_n), \tag{10.2}$$

where $\mathbf{W}(\mathbf{x};\delta_1) \circ \mathbf{W}(\mathbf{x};\delta_2) = \mathbf{W}(\mathbf{W}(\mathbf{x};\delta_2);\delta_1)$ denotes the composition of two warps. The warps are applied in a fine to coarse manner.

[1] The following is an elaboration on the paper [31] by Cootes *et al*, but is included here for completeness.



(a) Before.



(b) After.

Figure 10.1: The displacement of a pixel is governed by the displacement of its surrounding grid nodes [116].

10.2.2 Image Registration using the Lucas-Kanade Algorithm

The goal of the image registration algorithm is to align a target and a input image such that the difference is minimized. This is quantified by the minimization of the sum of squared residuals,

$$\mathbf{F}(\mathbf{p}) = \sum_{\mathbf{x}} [T(\mathbf{x}) - I(\mathbf{W}(\mathbf{x};\mathbf{p}))]^2, \tag{10.3}$$

where I is the input image and T is the target image. This can be minimized using a Gauss-Newton optimization scheme [82],

$$F(\mathbf{p}) = \frac{1}{2} \sum_{\mathbf{x}} \left[T(\mathbf{x}) - I(\mathbf{W}(\mathbf{x};\mathbf{p}+\Delta\mathbf{p}))\right]^2, \tag{10.4}$$

which by Taylor expansion and solving for $\boldsymbol{\Delta}\mathbf{p}$ gives,

$$\Delta\mathbf{p} = \mathbf{H}^{-1} \sum_{\mathbf{x}} \left[\nabla I(\mathbf{W}(\mathbf{x};\mathbf{p})) \frac{\partial \mathbf{W}(\mathbf{x};\mathbf{p})}{\partial \mathbf{p}}\right]^\top E(\mathbf{x}), \tag{10.5}$$

where $\mathbf{H}$ is the Gauss-Newton approximation to the Hessian,

$$\mathbf{H} = \sum_{\mathbf{x}} \left[\nabla I(\mathbf{W}(\mathbf{x};\mathbf{p})) \frac{\partial \mathbf{W}(\mathbf{x};\mathbf{p})}{\partial \mathbf{p}}\right]^\top \left[\nabla I(\mathbf{W}(\mathbf{x};\mathbf{p})) \frac{\partial \mathbf{W}(\mathbf{x};\mathbf{p})}{\partial \mathbf{p}}\right], \tag{10.6}$$

and the error is,

$$E(\mathbf{x}) = T(\mathbf{x}) - I(\mathbf{W}(\mathbf{x};\mathbf{p})). \tag{10.7}$$

The warp parameters $\mathbf{p}$ are updated using,

$$\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}. \tag{10.8}$$

The Jacobian is found to be,

$$\mathbf{J} = \sum_{\mathbf{x}} \left[\nabla I(\mathbf{W}(\mathbf{x};\mathbf{p})) \frac{\partial \mathbf{W}(\mathbf{x};\mathbf{p})}{\partial \mathbf{p}}\right], \tag{10.9}$$

where $\nabla I(\mathbf{W}(\mathbf{x};\mathbf{p}))$ is the image gradient of the input image sampled at the points $\mathbf{W}(\mathbf{x};\mathbf{p})$, and $\frac{\partial \mathbf{W}(\mathbf{x};\mathbf{p})}{\partial \mathbf{p}}$ is the derivative of the warp function with respect to the parameters.

This optimization scheme requires computation of the Jacobian $\mathbf{J}$ and the inverse Hessian $\mathbf{H}^{-1}$ at each iteration. For large volumes and large grids, this is very computationally demanding. However, the Hessian is symmetric and very sparse thus enabling the utilization of fast schemes for solving large sparse linear equations [47].

10.2.3 Inverse Compositional Image Registration

To overcome the drawbacks of the Gauss-Newton scheme of calculating the Jacobian and the Hessian in each iteration, Baker and Matthews [7] recently proposed the *Inverse Compositional Algorithm*, in which the Jacobian and the Hessian can be precomputed. As the name implies the algorithm consists of two innovations. The compositional part refers to the updating of the parameters and the inverse part indicates that the image and the target switches roles. The cost function in 10.4 is changed to,

$$F_{ic}(\mathbf{p}) = \frac{1}{2}\sum_{\mathbf{x}}\left[T(\mathbf{W}(\mathbf{x};\Delta\mathbf{p})) - I(\mathbf{W}(\mathbf{x};\mathbf{p}))\right]^2. \tag{10.10}$$

Solving for $\Delta\mathbf{p}$ gives,

$$\Delta\mathbf{p} = -\mathbf{H}_{ic}^{-1}\sum_{\mathbf{x}}\left[\nabla T(\mathbf{x})\frac{\partial\mathbf{W}(\mathbf{x};\mathbf{0})}{\partial\mathbf{p}}\right]^{\top} E(\mathbf{x}). \tag{10.11}$$

The update to the warp is,

$$\mathbf{W}(\mathbf{x};\mathbf{p}) = \mathbf{W}(\mathbf{x};\mathbf{p}) \circ \mathbf{W}(\mathbf{x};\Delta\mathbf{p})^{-1}, \tag{10.12}$$

In equation 10.10 it can be seen that the incremental warp $\mathbf{W}(\mathbf{x};\Delta\mathbf{p})$ applies only to the target T, and thus the Taylor expansion is around $\mathbf{p} = \mathbf{0}$, yielding the Jacobian

$$\mathbf{J}_{ic} = \sum_{\mathbf{x}}\left[\nabla T(\mathbf{W}(\mathbf{x};\mathbf{0}))\frac{\partial\mathbf{W}(\mathbf{x};\mathbf{0})}{\partial\mathbf{p}}\right]. \tag{10.13}$$

and thus the Hessian $\mathbf{H}_{ic}$ is,

$$\mathbf{H}_{ic} = \sum_{\mathbf{x}}\left[\nabla T(\mathbf{x})\frac{\partial\mathbf{W}(\mathbf{x};\mathbf{0})}{\partial\mathbf{p}}\right]^{\top}\left[\nabla T(\mathbf{x})\frac{\partial\mathbf{W}(\mathbf{x};\mathbf{0})}{\partial\mathbf{p}}\right]. \tag{10.14}$$

The Jacobian is independent of $\mathbf{p}$ and $\nabla T(\mathbf{x})$ is the image gradient of the target, thus enabling precomputation of the Jacobian and the Hessian.

Baker and Matthews [7] proves that the update $\Delta\mathbf{p}$ calculated using the inverse compositional algorithm is equivalent, to a first order approximation, to the update calculated using the Lucas-Kanade algorithm.

10.2.4 Adding Constraints

Baker *et al.* [8] describe how to incorporate prior information on the warp parameters. This could for instance be landmark or volume constraints formulated

as an additional term in the expression to be minimized, i.e. as weighted soft constraints to equation 10.3,

$$\frac{1}{2}\sum_{\mathbf{x}}[T(\mathbf{x}) - I(\mathbf{W}(\mathbf{x};\mathbf{p}))]^2 + \alpha\sum_{i=1}^{K} C_i^2(\mathbf{p}). \tag{10.15}$$

K is the number of constraints, C_i is a vector of functions containing the prior on the parameters for the i^{th} constraint and α is a weight controlling the emphasis on the prior term. In the inverse compositional framework this corresponds to equation 10.10,

$$\frac{1}{2}\sum_{\mathbf{x}}[T(\mathbf{W}(\mathbf{x};\Delta\mathbf{p}) - I(\mathbf{W}(\mathbf{x};\mathbf{p}))]^2 + \alpha\sum_{i=1}^{K} C_i^2(\mathbf{p} + \frac{\partial\mathbf{p}'}{\partial\Delta\mathbf{p}}\Delta\mathbf{p}). \tag{10.16}$$

Approximating this with a first order Taylor expansion gives the following update equations for the gradient $\Delta\mathbf{p}$ and the Hessian,

$$\begin{aligned}\Delta\mathbf{p} &= -\mathbf{H}_{ic,C_i}^{-1}\left[\sum_{\mathbf{x}}\left[\nabla T(\mathbf{x})\frac{\partial\mathbf{W}(\mathbf{x};\mathbf{0})}{\partial\mathbf{p}}\right]^{\top} E(\mathbf{x})\right. \\ &+ \left.\alpha_{Gr}\sum_{i=1}^{K}\left[\frac{\partial C_i}{\partial\mathbf{p}}\frac{\partial\mathbf{p}'}{\partial\Delta\mathbf{p}}\right]^{T} C_i(\mathbf{p})\right] \end{aligned} \tag{10.17}$$

$$\mathbf{H}_{ic,C_i} = \mathbf{H}_{ic} + \alpha_{He}\sum_{i=1}^{K}\left[\frac{\partial C_i}{\partial\mathbf{p}}\frac{\partial\mathbf{p}'}{\partial\Delta\mathbf{p}}\right]^{T}\left[\frac{\partial C_i}{\partial\mathbf{p}}\frac{\partial\mathbf{p}'}{\partial\Delta\mathbf{p}}\right]. \tag{10.18}$$

The computational cost of adding priors is that the Hessian is not constant anymore. The costs is $O(nN + n^2K + n^3)$ compared to $O(nN + n^2)$ without priors. As long as the number of constraints and the number of parameters are smaller than the number of pixels/voxels ($K << N$ and $n << N$), this cost is negligible. In order to make the prior terms robust to the number of constraints, the α weights are chosen relative to the L2-norm of the term without priors,

$$\alpha_{Gr} = \alpha_{Rel}\frac{\left\|\sum_{\mathbf{x}}\left[\nabla T(\mathbf{x})\frac{\partial\mathbf{W}(\mathbf{x};\mathbf{0})}{\partial\mathbf{p}}\right]^{T} E(\mathbf{x})\right\|_{L2}}{\left\|\sum_{i=1}^{K}\left[\frac{\partial C_i}{\partial\mathbf{p}}\frac{\partial\mathbf{p}'}{\partial\Delta\mathbf{p}}\right]^{T} C_i(\mathbf{p})\right\|_{L2}} \tag{10.19}$$

$$\alpha_{He} = \alpha_{Rel}\frac{\left\|\mathbf{H}_{ic}\right\|_{L2}}{\left\|\sum_{i=1}^{K}\left[\frac{\partial C_i}{\partial\mathbf{p}}\frac{\partial\mathbf{p}'}{\partial\Delta\mathbf{p}}\right]^{T}\left[\frac{\partial C_i}{\partial\mathbf{p}}\frac{\partial\mathbf{p}'}{\partial\Delta\mathbf{p}}\right]\right\|_{L2}}, \tag{10.20}$$

where $\alpha_{Rel} >= 0$ is the relative weighting between the two terms. $\alpha_{Rel} = 0$ corresponds to the inverse compositional without the prior term.

10.2.5 Fast Grid-Based Image Registration

The inverse compositional scheme described above requires the computation the Jacobian from equation 10.13 and subsequently the Hessian from equation 10.14. To take full advantage of the inverse compositional method, they must be precalculated and stored. However, the storage requirement of the Jacobian can be very large. In a naive implementation using a simple $3 \times 3 \times 3$ grid, space is needed for $81 \cdot N$ floating point numbers with N being the number of pixels in the image and potentially very large. So great care must be taken in the implementation in order to exploit the speed gain provided by the inverse compositional framework.

10.2.5.1 Calculating the Jacobian and the Hessian

The Jacobian consist of the gradient $\nabla T(\mathbf{x}) = \left[\frac{\partial T}{\partial x} \quad \frac{\partial T}{\partial y} \quad \frac{\partial T}{\partial z}\right]$ of the target image and the derivative of the warp $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$. $\nabla T(\mathbf{x})$ can be found using a simple finite difference method or more elaborate methods using B-splines or similar interpolating methods. The warp derivatives are very simple to calculate. As mentioned, the parameters of the warp are simply the $3 \cdot mnp$ ordinates of the displacements of the nodes,

$$\mathbf{p} = \left[d^x_{1,1,1}\ d^y_{1,1,1}\ d^z_{1,1,1}\ d^x_{2,1,1}\ d^y_{2,1,1}\ d^z_{2,1,1} \cdots\ d^x_{m,n,p}\ d^y_{m,n,p}\ d^z_{m,n,p}\right], \tag{10.21}$$

for a $m \times n \times p$ grid. This and equation 10.1 yields

$$\frac{\partial \mathbf{W}}{\partial d^x_{i,j,l}} = \frac{\partial \mathbf{W}}{\partial d^y_{i,j,l}} = \frac{\partial \mathbf{W}}{\partial d^z_{i,j,l}}, \tag{10.22}$$

meaning that the derivatives corresponding to the x, y, z components of one displacement vector are equal. Each pixel $\mathbf{x}$ contributes to eight partial derivatives only, corresponding to the eight surrounding grid nodes, cf. figure 10.2 for a 2D example. Thus, for N pixels, an $N \times 8$ floating point value representation of $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ is possible using a simple lookup method. However, since the kernel function $k()$ only operates on the distance from a contributing pixel $\mathbf{x}$ to a node, an even sparser representation of only $N_{reg} \times 8$ is possible. N_{reg} is the number of pixels surrounded by 8 neighboring grid nodes. For increasing grid sizes the space requirement goes down.

Another property of $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ is seen from,

$$k(i + 1 - x) = 1 - k(x - i), \tag{10.23}$$

which holds for y and z as well. For a neighborhood of grid nodes, the derivative

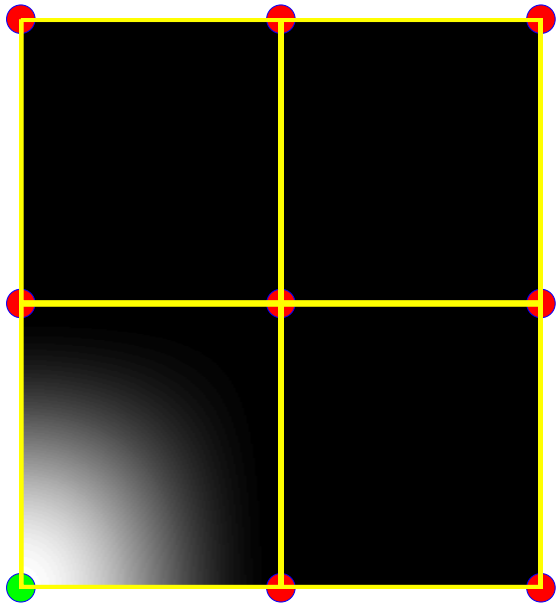


Figure 10.2: The derivative of the warp with respect to the node in the lower left corner. It is non-zero only in the region bounded by the neighboring nodes.

at $\mathbf{x} = (x, y, z)$ wrt. the parameters of the warp has the form,

$$\begin{aligned}
\frac{\partial \mathbf{W}}{\partial p_{i,j,l}} &= k(x-i)k(y-j)k(z-l) \\
\frac{\partial \mathbf{W}}{\partial p_{i+1,j,l}} &= (1-k(x-i))k(y-j)k(z-l) \\
&\vdots \\
\frac{\partial \mathbf{W}}{\partial p_{i+1,j+1,l+1}} &= (1-k(x-i))(1-k(y-j))(1-k(z-l)) \qquad (10.24)
\end{aligned}$$

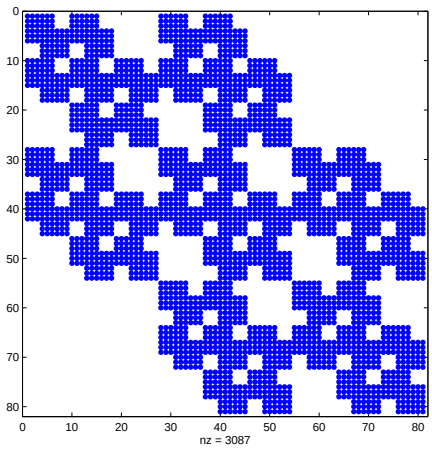
Thus, to evaluate the derivative contribution for a single pixel $\mathbf{x}$, only three kernel function evaluations, $k(x-i)$, $k(y-j)$ and $k(z-l)$ are needed. The derivatives can then be found using simple multiplications and subtractions.

Finally the Jacobian is calculated as in equation 10.13, which means multiplying the warp derivatives with the image gradients. However, the Jacobian with the sparse representation mentioned above has a size of $N \times 24$ floating point values which requires a large amount of memory. Subsequently a compromise has to be made, and in this work the image gradients and the warp derivatives are stored separately, yielding space requirement for $N \times 3$ plus $N_{reg} \times 8$ numbers. This means the multiplication of the gradients and the derivatives must be performed each time the Jacobian is needed, yet precalculation of the Hessian is still possible.

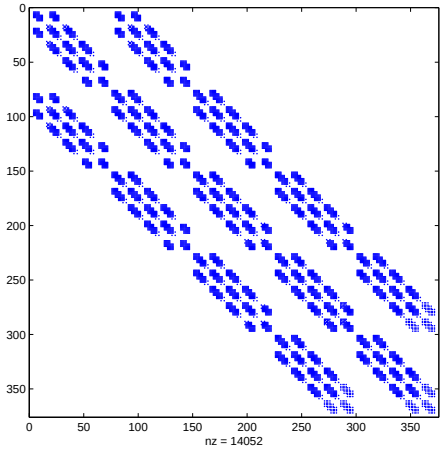
The Hessian is calculated as in equation 10.14. The sparseness of the Jacobian is transferred into the Hessian. Figure 10.3 depicts the sparseness of the Hessian for three grid sizes. Solving equation 10.11 requires the inversion of the Hessian matrix. This usually destroys the sparseness and is very computationally demanding. Furthermore, the Hessian can be very ill-conditioned, but making use of iterative methods [47] for solving sparse linear equations, the inversion of the Hessian can be avoided.

10.2.5.2 The Image Registration Algorithm

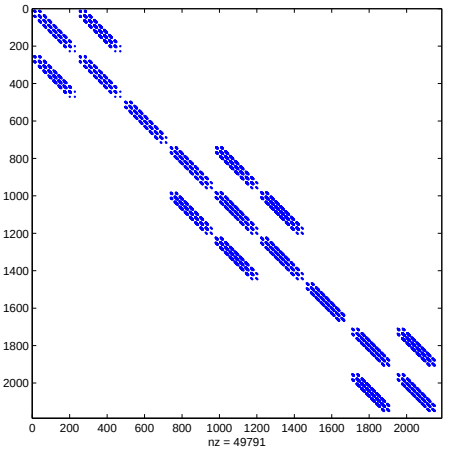
To estimate the transformation of the input image I into the target image T minimization of equation 10.10 with respect to $\mathbf{p}$ is required. This is done by first applying a coarse grid, eg. $3 \times 3 \times 3$, and iteratively solving equation 10.11 until convergence. Subsequently a finer grid, eg. $5 \times 5 \times 5$, is applied, and so forth. This enables the estimation of small local transformations while still being diffeomorphic [31]. A multilevel approach, using downsampled versions of the images, is adopted to avoid local minima. Consequently, for each grid size, optimization is done in a coarse-to-fine manner as well, starting the optimization



(a) 81×81 matrix.



(b) 375×375 matrix.



(c) 2187×2187 matrix.

Figure 10.3: Plot of non-zero elements in Hessian matrices, for grid sizes of $3 \times 3 \times 3$, $5 \times 5 \times 5$ and $9 \times 9 \times 9$.

on a finer downsampling level with the parameters estimated on a coarser level. See figure 10.4. At each iteration equation 10.12 must be used to update the parameters. In this work the parameters of $\mathbf{W}(\mathbf{x};\mathbf{\Delta p})$ is estimated with the first order approximation [8] $-\mathbf{\Delta p}$. Thus the parameter update from equation 10.12 has the form,

$$\mathbf{p} \leftarrow \mathbf{p} - \mathbf{\Delta p}. \tag{10.25}$$

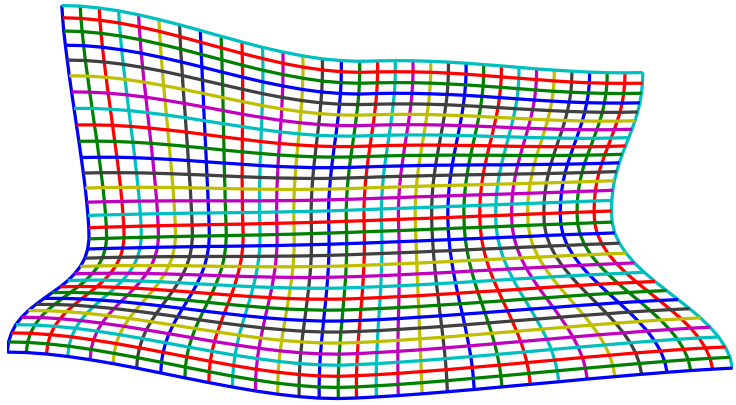
When the optimization for one grid size converges, a finer grid is applied, ie. a $3 \times 3 \times 3$ is replaced with a $5 \times 5 \times 5$. Composition of warps of difference grid sizes is done in a simple manner. Figure 10.5 depicts the scheme. The grid nodes of the higher level warp are transformed with the lower level warp. Thus, the parameters from the lower level are transported into the higher level warp.

10.3 Results

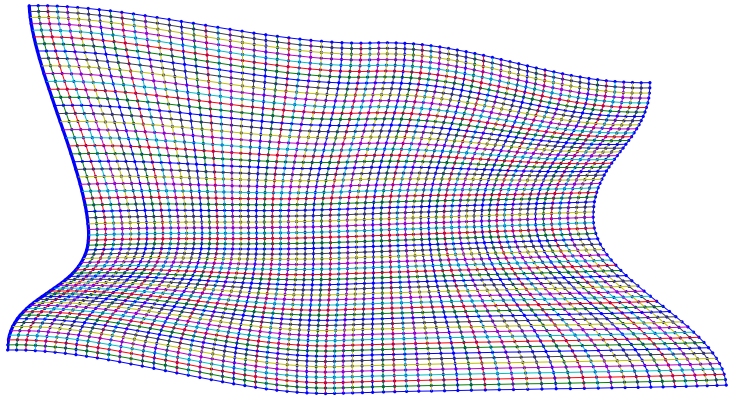
10.3.1 3D Non-Rigid Registration

5 CT-volumes of the hind part of porcine carcasses are cross-registered to compare the inverse compositional algorithm with the Lucas-Kanade algorithm. The volumes are approximately 512x512x170 voxels of size [0.67,0.67,2] mm. After rigid registration the two algorithms for non-rigid registration are applied and the speed and accuracy are compared. Due to time considerations the 20 registrations are done with images downsampled to $\frac{1}{16}$, $\frac{1}{8}$ and $\frac{1}{4}$ of the original image size for grids of size [3,5,7,9] per dimension. Figure 10.6 shows three slice planes of a volume and their corresponding error images after a typical registration. The main errors are along the border of the volume due to the large difference in value between background and volume.

Rows 1 and 2 in table 10.1 show the mean value and standard deviation of the number of iterations used before convergence, the final registration error and the time consumption for the two algorithms. Row 3 shows the mean improvement when using the IC algorithm and row 4 shows a paired T-test of significant differences in the mean values. There are highly significant improvements (denoted by 1) in both speed and number of iterations. The mean final registration error of the two methods are not significantly different. In this simple test it therefore shows that the inverse compositional algorithm is as accurate as the Lucas-Kanade algorithm, as expected, but is twice as fast for registration of CT-volumes. The results are obtained using a Dell Latitude 810D, with a 2.0Ghz CPU and 2 gb of ram. The implementation of the Lucas-Kanade algorithm utilizes the same sparseness properties as the implementation of the inverse compositional algorithm. Thus, the algorithms perform similar memory wise. The speed increase of the inverse compositional algorithm is due to the



(a) Coarser level.

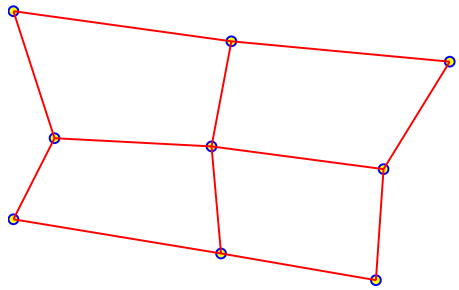


(b) Finer level.

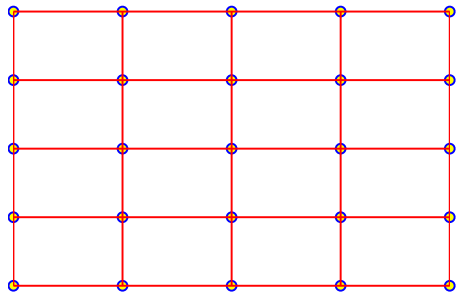
Figure 10.4: Plot of the warpfield estimated on a coarse image level and the corresponding field in a finer level.

Table 10.1: Comparison between the Lucas-Kanade algorithm and the inverse compositional algorithm based on 20 registrations.

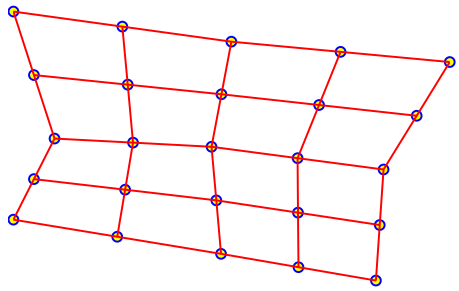
	Iterations	Final Error	Speed [sec.]
Lucas-Kanade (mean±std.)	160 ± 29	$(7.6 \pm 4.1) \cdot 10^{11}$	436 ± 128
Inverse Compositional (mean±std.)	140 ± 15	$(7.5 \pm 4.0) \cdot 10^{11}$	222 ± 45
Mean improvement with IC (%)	13	2	49
Paired T-test for difference in mean, 1=mean values are sign. diff.	1 ($p < 0.006$)	0	1 ($p < 10^{-3}$)



(a) Resulting 3×3 grid.

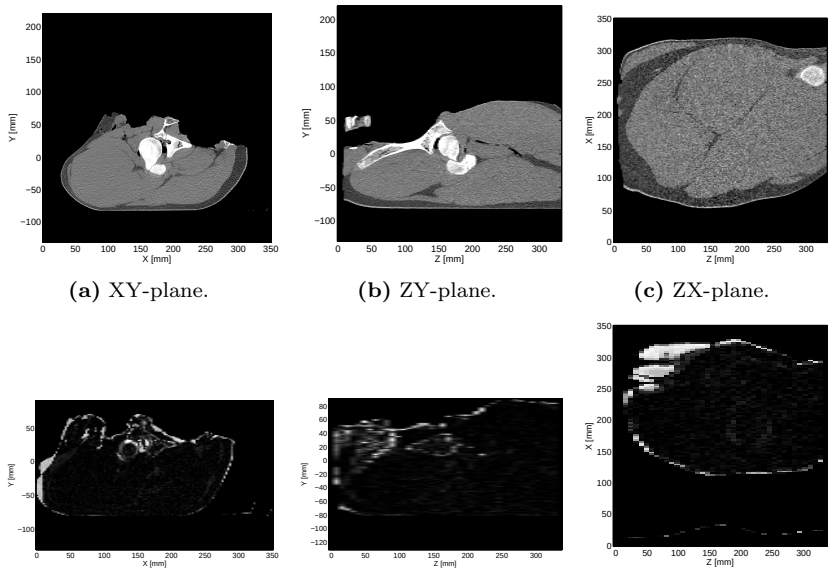


(b) New 5×5 grid



(c) Transformed 5×5 grid

Figure 10.5: Composition of grids of different sizes.



(a) XY-plane. **(b)** ZY-plane. **(c)** ZX-plane.

(d) XY error image. **(e)** ZY error image. **(f)** ZX error image.

Figure 10.6: (a)-(c): CT-volume of a porcine carcass. (d)-(f): Error images.

precomputation of the Hessian and the target image gradient. In the paper [31] by Cootes *et al.* the optimization is done by a simple gradient descent scheme. The gradient is computed by displacement of each of the grid nodes in turn. Early experiments have shown this approach to be very slow, so no comparison is made. However, no space is required for precomputation, and as such it performs well memory wise.

10.3.2 Adding Constraints

If some prior knowledge is at hand before registration or if there is a need for guiding the registration, adding constraints in the optimization scheme should be considered. As an example 58 landmark constraints are applied in a 2D affine warp of image I in figure 10.7(a) to image T in figure 10.7(b). The warp $\mathbf{W}(\mathbf{x}, \mathbf{p})$ of a pixel $\mathbf{x}$ is defined by 6 parameters in vector $\mathbf{p}$

$$\mathbf{W}(\mathbf{x}, \mathbf{p}) = \begin{pmatrix} 1+p_1 & p_3 & p_5 \\ p_2 & 1+p_4 & p_6 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \tag{10.26}$$

The first order term in the linearization of the additive update is

$$\frac{\partial \mathbf{p}'}{\partial \Delta \mathbf{p}} = -\begin{pmatrix} 1+p_1 & p_3 & 0 & 0 & 0 & 0 \\ p_2 & 1+p_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1+p_1 & p_3 & 0 & 0 \\ 0 & 0 & p_2 & 1+p_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1+p_1 & p_3 \\ 0 & 0 & 0 & 0 & p_2 & 1+p_4 \end{pmatrix}. \quad (10.27)$$

The error function of the prior part is defined as

$$C_i = (Lm_{T,i} - \mathbf{W}(Lm_{I,i}; \mathbf{p})), \quad (10.28)$$

where $Lm_{T,i}$ is the i^{th} landmark in the target image T and $Lm_{I,i}$ is the i^{th} landmark in the input image I. The Jacobian of C_i then is

$$\frac{\partial C_i}{\partial \mathbf{p}} = -\begin{pmatrix} Lm_{I,i,x} & 0 & Lm_{I,i,y} & 0 & 1 & 0 \\ 0 & Lm_{I,i,x} & 0 & Lm_{I,i,y} & 0 & 1 \end{pmatrix}, \quad (10.29)$$

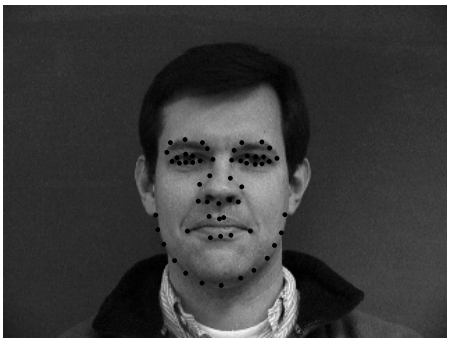
where $Lm_{I,i,x}$ and $Lm_{I,i,y}$ are the x- and y-coordinates, respectively, of the i^{th} landmark in I.

Figure 10.7(c) shows the L2-norm of the intensity and prior error as α_{Rel} is increased. α_{Rel}=0 corresponds to no constraints and α_{Rel}=1 corresponds to the intensity and the constraints being weighted equally. The constraints improve the registration as long as the intensity error is decreasing, i.e. for α-values approximately between 0.8 and 1.6. The prior error will off course decrease with increasing α-value. How much weight to put on the constraints depends on the application but for this example weighting intensity and prior more or less equal gives the best result. Figures 10.8(a)-10.8(c) show the difference between the input image I and the target image T warped into the coordinate frame of I, for α_{Rel}=0,1 and 2. The improvement in registration without constraints compared to the registration with the intensity and constraints weighted equally is obvious, especially in the area around the jaw.

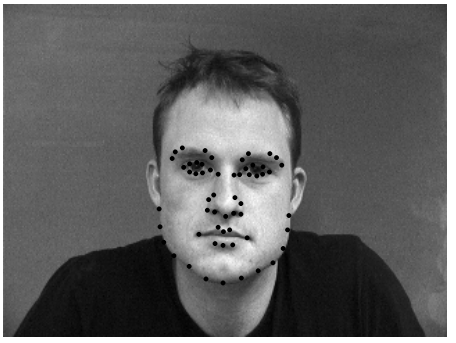
Applying similar constraints as described in section 10.2.4 to the nonrigid case or in 3D is straightforward, all you need to do is to define C_i and compute the Jacobian and $\partial \mathbf{p}'/\partial \Delta \mathbf{p}$.

10.4 Conclusion

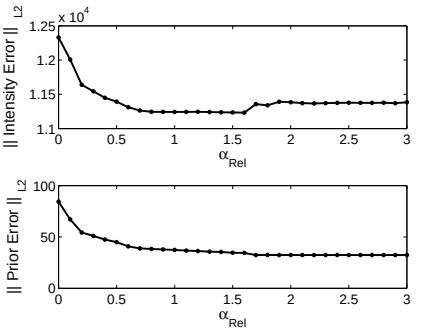
This paper has presented an algorithm for registration of 3D images. Registration is done using grid-based warps in a coarse-to-fine manner, enabling the



(a) Input image I.

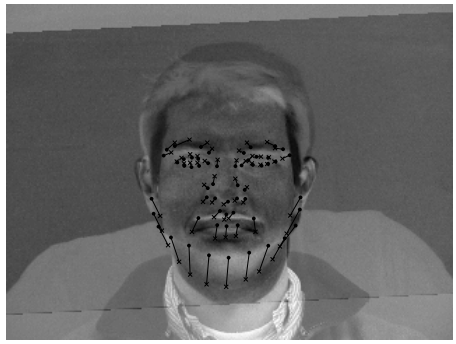


(b) Target image T.

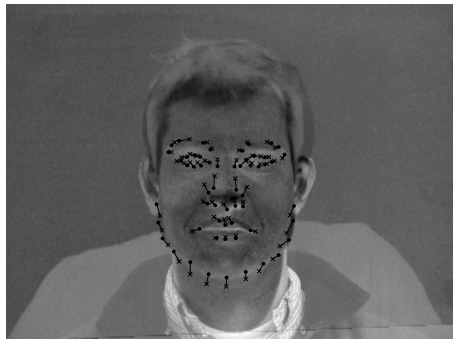


(c) Intensity and prior errors vs. α_{Rel}.

Figure 10.7: 2D affine registration of 2 images with increasing weight on landmark constraints.



(a) $I(\mathbf{x}) - T(\mathbf{W}(\mathbf{x};\mathbf{p})^{-1})$, α_{Rel}=0.



(b) $I(\mathbf{x}) - T(\mathbf{W}(\mathbf{x};\mathbf{p})^{-1})$, α_{Rel}=1.

(c) $I(\mathbf{x}) - T(\mathbf{W}(\mathbf{x};\mathbf{p})^{-1})$, α_{Rel}=2.

Figure 10.8: 2D affine registration of 2 images with increasing weight on landmark constraints.

registration of even fine structures in the images while still being diffeomorphic. Using the inverse compositional framework for optimization, the algorithm performs very fast. Exploitation of the sparseness of the grid-based warps and the properties of the interpolating kernel, enables the precomputation of the Hessian and the target image gradient. The algorithm has a two-fold increase in speed compared to a Lucas-Kanade based algorithm.

Acknowledgments

The 2D images are part of the DTU Face Database [116] and the CT data was provided by the Danish Meat Research Institute as a part of the project "The Virtual Butcher" funded by the Danish Pig Levy Fund and the Directorate for Food, Fisheries and Agri Business. The authors would also like to thank the IMM ITMAN graduate school at the Technical University of Denmark for support.

Appendix A

Demos

This appendix contain examples of two demo-tools produced during the course of the project. Both where implemented in order to demonstrate the concepts behind the Virtual Slaughterhouse to employees in the Danish slaughterhouse industry.

A.1 OinkExplorer

Live demonstration of the concept of virtual cuts. A tool for interactive placement of a cutting plane. Real time estimated of the weight of each product is displayed while the plane is moved.

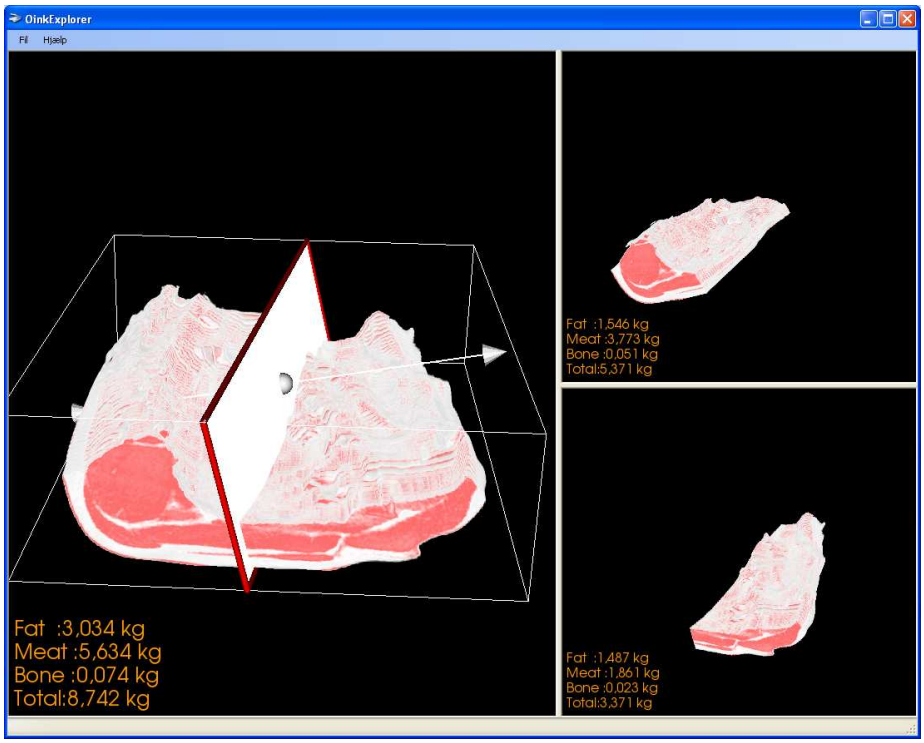


Figure A.1: Demo application: OinkExplorer. Demonstrating interactive placement of a cutting plane.

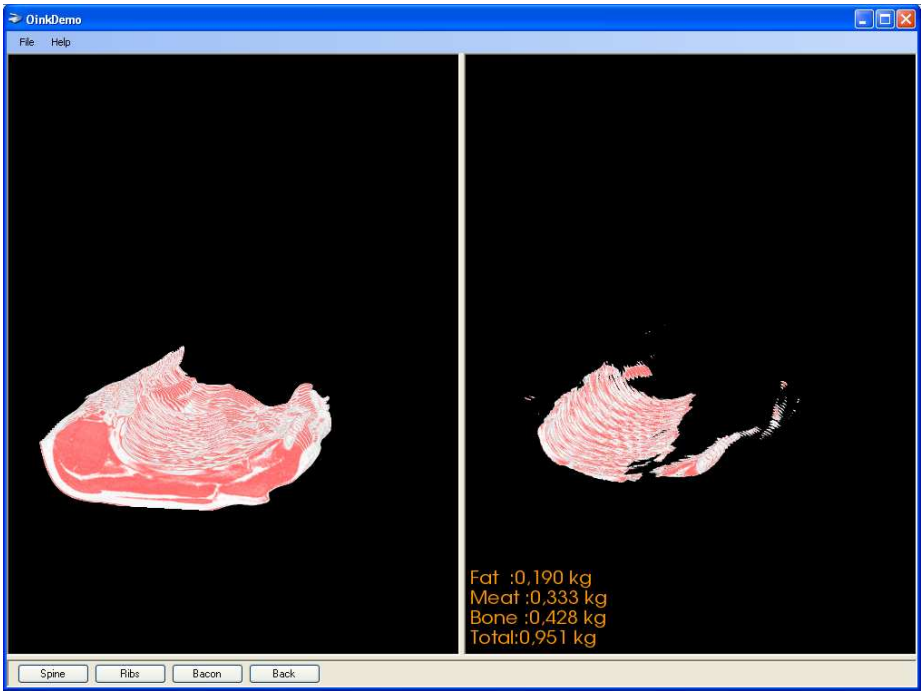


Figure A.2: Demo application: OinkDemo. Cuts a middle part into of a carcass into ribs, 18cm back and belly.

A.2 OinkDemo

Demonstration of automatic cutting of a middle part of a carcass into ribs, 18cm back and belly. Fat, meat and bone content is displayed.

List of Tables

List of Figures

List of Algorithms

Bibliography

[1] E.H. Adelson, C.H. Anderson, J.R. Bergen, P.J. Burt, and J.M. Ogden. Pyramid methods in image processing. *RCA Engineer*, 29(6):33–41, 1984.

[2] Canadian Food Inspection Agency. Meat cuts manual, 2004. URL http://www.inspection.gc.ca/english/fssa/labeti/mcmancv/mcmancve.shtml.

[3] P. Allen and O. Vangen. X-ray tomography of pigs: some preliminary results. *In Vivo Measurement of Body Composition in Meat Animals*, pages 52–66, 1984.

[4] J. Ashburner. *Computational Neuroanatomy*. PhD thesis, University of London, 2001.

[5] J. Ashburner and K.J. Friston. Nonlinear Spatial Normalization Using Basis Functions. *Human Brain Mapping*, 7:254–266, 1999.

[6] R. Bajcsy and S. Kovačič. Multiresolution elastic matching. *Computer Vision, Graphics, and Image Processing*, 46(1):1–21, 1989.

[7] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56(3):221–255, February 2004.

[8] S. Baker, R. Gross, and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework: Part 4. Technical report, Robotics Institute, Carnegie Mellon University, 2004.

[9] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. SIAM, Philadelphia, PA, 1994.

[10] K. Bhatia, P. Aljabar, J. Boardman, L. Srinivasan, M. Murgasova, S. Counsell, M. Rutherford, J. Hajnal, A. Edwards, and D. Rueckert. Groupwise combined segmentation and registration for atlas construction. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2007*, pages 532–540. Springer, 2007.

[11] J.M. Bland and D.G. Altman. Statistical methods for assessing agreement between two methods of clinical measurement. *Lancet*, 1(8476):307–310, 1986.

[12] D. Blezek and J. Miller. Atlas stratification. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2006*, pages 712–719. Springer, 2006.

[13] F.L. Bookstein. Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, 1989.

[14] R. Brandt, T. Rohlfing, J. Rybak, S. Krofczik, A. Maye, M. Westerhoff, HC Hege, and R. Menzel. Three-dimensional average-shape atlas of the honeybee brain and its applications. *Journal of Comparative Neurology*, 492(1):1–19, 2005.

[15] M. Bro-Nielsen. *Medical Image Registration and Surgery Simulation*. PhD thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 1996.

[16] M. Bro-Nielsen and C. Gramkow. Fast fluid registration of medical images. In *Proc. Visualization in Biomedical Computing*, pages 267–276, 1996.

[17] C. Broit. *Optimal registration of deformed images*. PhD thesis, University of Pennsylvania, 1981.

[18] Lisa Gottesfeld Brown. A survey of image registration techniques. *ACM Comput. Surv.*, 24(4):325–376, 1992. ISSN 0360-0300.

[19] Paul E. Buis and Wayne R. Dyksen. Efficient vector and parallel manipulation of tensor products. *ACM Trans. Math. Softw.*, 22(1):18–23, 1996. ISSN 0098-3500. doi: http://doi.acm.org/10.1145/225545.225548.

[20] M. Chen. *3-D Deformable Registration Using a Statistical Atlas with Applications in Medicine*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, October 1999.

[21] Z.-H. Cho, J. P. Jones, and M. Singh. *Foundations of Medical Imaging*. John Wiley & Sons, Inc., 1993.

[22] G. Christensen. *Deformable Shape Models for Anatomy*. PhD thesis, Washington University, 1994.

[23] L. B. Christensen, M. Vester-Christensen, C. Borggaard, and E. V. Olsen. Robustness of weight and meat content in pigs determined by CT. In *54th International Congress of Meat Science and Technology (ICoMST)*, 2008.

[24] L.B. Christensen and C. Borggaard. Challenges in the approval of ct as future reference for grading of farmed animals. In *51st International Congress of Meat Science and Technology (ICoMST)*, 2005.

[25] G. Collewet, P. Bogner, P. Allen, H. Busk, A. Dobrowolski, E. Olsen, and A. Davenel. Determination of the lean meat percentage of pig carcasses using magnetic resonance imaging. *Meat Science*, 70:563–572, 2005.

[26] D. L. Collins and A. C. Evans. Animal: validation and applications of non-linear registration-based segmentation. *International Journal and Pattern Recognition and Artificial Intelligence*, 11:1271–1294, 1997.

[27] Commission of the European Communities (EC). EC regulation no. 3127/94 amending regulation (EC) no. 2967/85 laying down detailed rules for the application of the community scale for grading pig carcasses. ECOJ Nr L330, 43, 1994.

[28] The Council Of The European Communities. Council regulation (eec) no 3220/84 of 13 november 1984 determining the community scale for grading pig carcases. Technical report, 1984.

[29] T. F. Cootes and Taylor. Active shape models-'smart snakes'. In *Proc. British Machine Vision Conf., BMVC92*, pages 266–275, 1992.

[30] T.F. Cootes, S. Marsland, C.J. Twining, K. Smith, and C.J. Taylor. Groupwise diffeomorphic non-rigid registration for automatic model building. In *European Conference on Computer Vision*, pages Vol IV: 316–327, 2004.

[31] T.F. Cootes, C.J. Twining, and C.J. Taylor. Diffeomorphic statistical shape models. In *British Machine Vision Conference*, pages 447–456, 2004.

[32] T.F. Cootes, C.J. Twining, K.O.Babalola, and C.J. Taylor. Diffeomorphic statistical shape models. *Image and Vision Computing*, 26(3):326–332, 2008.

[33] R. H. Davies, C. J. Twining, P. D. Allen, T. F. Cootes, and C. J. Taylor. Shape discrimination in the hippocampus using an mdl model. In *Information Processing in Medical Imaging*, Lecture Notes in Computer Science, 2003.

[34] A. Dobrowolski, W. Branscheid, R. Romvári, P. Horn, and P. Allen. X-ray computed tomography as possible reference for the pig carcass evaluation. *Fleischwirtschaft*, 84 (3):109–112, 2004.

[35] S. G. H. Erbou. *Modeling the Biological Diversity of Pig Carcasses*. PhD thesis, DTU, 2008.

[36] S. G. H. Erbou, S. Darkner, J. Fripp, S. Ourselin, and B. K. Ersbøll. Estimation of Shape Model Parameters for 3D Surfaces. In *IEEE 5th Int. Symposium on Biomedical Imaging (ISBI)*, pages 624–627, 2008.

[37] S. G. H. Erbou, M. Vester-Christensen, L. B. Christensen, R. Larsen, and B. K. Ersbøll. Sparse point distribution models. *Machine Vision and Applications*, Revised manuscript submitted August 2008, 2008.

[38] G6RD-CT-1999-00127 EUPIGCLASS. Eupigclass. www.eupigclass.net, 1999. Details to be found at http://www.eupigclass.net/.

[39] G. Fasano, F. Lampariello, and M. Sciandrone. A truncated nonmonotone gauss-newton method for large-scale nonlinear least-squares problems. *Comput. Optim. Appl.*, 34(3): 343–358, 2006. ISSN 0926-6003.

[40] B. Fischer and J. Modersitzki. Fast inversion of matrices arising in image processing. *Numerical Algorithms*, 22:1–11, 1999.

[41] B. Fischer and J. Modersitzki. Fast diffusion registration. *Inverse Problems, Image Analysis, and Medical Imaging: Ams Special Session on Interaction of Inverse Problems and Image Analysis, January 10-13, 2001, New Orleans, Louisiana*, 313, 2002.

[42] B. Fischer and J. Modersitzki. Curvature Based Image Registration. *Journal of Mathematical Imaging and Vision*, 18(1):81–85, 2003.

[43] P. E. Frandsen, K. Jonasson, H. B. Nielsen, and O. Tingleff. *Unconstrained Optimization, 3rd edition*. Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2004.

[44] T. Gaens, F. Maes, D. Vandermeulen, and P. Suetens. Non-rigid Multimodal Image Registration Using Mutual Information. In *Medical Image Computing and Computer-Assisted Interventation - MICCAI'98*, Lecture Notes in Computer Science, pages 1099–1106. Springer, 1998.

[45] J.C. Gee, D.R. Haynor, L. Le Briquer, and R. Bajcsy. Advances in elastic matching theory and its implementation. *Proceedings of the First Joint Conference on Computer Vision, Virtual Reality and Robotics in Medicine and Medial Robotics and Computer-Assisted Surgery*, pages 63–72, 1997.

[46] C. A. Glasbey and C. D. Robinson. Estimators of tissue proportions from X-ray CT images. *Biometrics*, 58:928–936, 2002.

[47] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, USA, second edition, 1989.

[48] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall, 2nd edition, 2002.

[49] A.A. Goshtasby. *2-D and 3-D Image Registration: for Medical, Remote Sensing, and Industrial Applications*. Wiley-Interscience, 2005.

[50] Anders Grønlund. Optimized raw material utilization. Master's thesis, Faculty of Life Sciences, University of Copenhagen, 2008.

[51] A. Guimond, J. Meunier, and J. Thirion. Average brain models: A convergence study. *Computer Vision and Image Understanding*, 77(2):192–210, 2000.

[52] Y. Guo. *Medical Image Registration and Application to Atlas-Based Segmentation*. PhD thesis, Kent State University, 2006.

[53] E. Haber and J. Modersitzki. A multilevel method for image registration. *SIAM J. Sci. Comput.*, 27(5):1594–1607, 2006.

[54] E. Haber and J. Modersitzki. Image Registration with Guaranteed Displacement Regularity. *International Journal of Computer Vision*, 71(3):361–372, 2007.

[55] J.V. Hajnal, D.J. Hawkes, and D.L.G. Hill. *Medical Image Registration*. CRC Press, 2001.

[56] M. F. Hansen. Quality estimation and segmentation of pig backs. Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2005.

[57] M. F. Hansen, S. Erbou, M. Vester-Christensen, R. Larsen, B. Ersbøll, and L. B. Christensen. Surface-to-Surface Registration Using Level Sets. In *Scandinavian Conference on Image Analysis (SCIA)*, pages 780–788, 2007.

[58] P.C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*. Society for Industrial Mathematics, 1998.

[59] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, and R. Tibshirani. *The elements of statistical learning*. Springer New York, 2001.

[60] P. Hellier, C. Barillot, E. Memin, and P. Perez. Medical image registration with robust multigrid techniques. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI'99*, Lecture Notes in Computer Science. Springer, 1999.

[61] P. Hellier, C. Barillot, I. Corouge, B. Gibaud, G. Le Goualher, D.L. Collins, A. Evans, G. Malandain, N. Ayache, G.E. Christensen, and H.J.; Johnson. Retrospective evaluation of intersubject brain registration. *Medical Imaging, IEEE Transactions on*, 22(9): 1120 – 1130, 2003.

[62] S. Henn and K. Witsch. Iterative multigrid regularization techniques for image matching. *SIAM Journal on Scientific Computing*, 23(4):1077–1093, July 2002. ISSN 1064-8275 (print), 1095-7197 (electronic).

[63] M.R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stand*, 49(6):409–436, 1952.

[64] P. W. Holland and R. E. Welsch. Robust regression using iteratively reweighted least-squares. *Communications in Statistics - Theory and Methods*, 6(9):813 – 827, 1977.

[65] C. Hvid. Grisen er på vores vinderhold. Feature in Jyllands-Posten, September 2008.

[66] M. Hviid and M. Vester-Christensen. Density of lean meat tissue in pork - measured by ct. In *54th International Congress of Meat Science and Technology - ICoMST*, 2008.

[67] S.A. Jackson, F.R.M. Thomas, and S.N. Harrison. *Cross-sectional Imaging Made Easy*. Churchill Livingstone, 2004.

[68] J. Johansen, B. Egelandsdal, M. Røe, K. Kvaal, and A. H. Aastveit. Calibration models for lamb carcass composition analysis using computerized tomography (CT) imaging. *Chemometrics and Intelligent Laboratory Systems*, 87(2):303–311, 2007.

[69] S. Joshi, B. Davis, M. Jomier, and G. Gerig. Unbiased diffeomorphic atlas construction for computational anatomy. *Neuroimage*, 23:151–160, 2004.

[70] W.A. Kalender. *Computed Tomography*. Publicis Corporate, second edition, 2005.

[71] N. C. Kjærsgaard. The value of improved measurements in a pig slaughterhouse. Technical report, Department of Management Engineering, Technical University of Denmark., 2008.

[72] N. C. Kjærsgaard. The value of a general increase in slaughter weight for pigs. Technical report, Department of Management Engineering, Technical University of Denmark., 2008.

[73] N. C. Kjærsgaard. Limitations in production and stock and their effect on the profitability of the slaughterhouses. Technical report, Department of Management Engineering, Technical University of Denmark., 2008c.

[74] N. C. Kjærsgaard. Evaluating different sorting criteria and strategies using mathematical programming. Technical report, Department of Management Engineering, Technical University of Denmark, 2008.

[75] N. C. Kjærsgaard. *Optimization of the raw material use at Danish slaughterhouses*. PhD thesis, DTU informatics, Technical University of Denmark, 2008.

[76] J. Kongsro, M. Røe, K. Kvaal, AH Aastveit, and B. Egelandsdal. Prediction of fat, muscle and value in Norwegian lamb carcasses using EUROP classification, carcass shape and length measurements, visible light reflectance and computer tomography (CT). *Meat Science*, 2008.

[77] J. Kybic. *Elastic Image Registration Using Parametric Deformation Models*. EPFL thesis no. 2439 (2001), 180 p., Swiss Federal Institute of Technology Lausanne (EPFL), July 26, 2001.

[78] R. Larsen. 3-D contextual Bayesian classifiers. *IEEE Transactions on Image Processing*, 10(3):518–524, 2000.

[79] S. Lee, G. Wolberg, and S.Y. Shin. Scattered data interpolation with multilevel b-splines. *Visualization and Computer Graphics, IEEE Transactions on*, 3(3):228 – 244, 1997.

[80] L. Lemieux, Kitchen N. D., Hughes, Stephen W., and David G. T. Thomas. Voxel-based localization in frame-based and frameless stereotaxy and its accuracy. *Medical Physics*, 21:1301–1310, 1994.

[81] H. Lester and S.R. Arridge. A survey of hierarchical non-linear medical image registration. *Pattern Recognition*, 32(1):129–149, 1999.

[82] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision (darpa). In *Proceedings of the 1981 DARPA Image Understanding Workshop*, pages 121–130, April 1981.

[83] A. Lyckegaard. Modelling of cutting of pig carcasses. Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2007.

[84] A. Lyckegaard, R. Larsen, L. B. Christensen, M. Vester-Christensen, and E. V. Olsen. Contextual analysis of CT scanned pig carcasses. In *52nd International Congress of Meat Science and Technology (ICoMST)*, pages 207–208, 2006.

[85] F. Maes, A. Collignon, D. Vandermeulen, P. Suetens, and G. Marchal. Multimodality image registration by maximization of mutual information. *IEEE Trans. Medical Imaging*, 16(2):187–198, April 1997.

[86] J.B.A. Maintz and M.A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):1–16, 1998.

[87] J. Modersitzki. *Numerical Methods for Image Registration*. Numerical Mathematics and Scientific Computation. Oxford University Press, 2004.

[88] M. Monziols, G. Collewet, F. Mariette, M. Kouba, and A. Davenel. Muscle and fat quantification in MRI gradient echo images using a partial volume detection method. Application to the characterization of pig belly tissue. *Magnetic Resonance Imaging*, 23(6):745–755, 2005.

[89] M. Monziolsa, G. Colleweta, M. Bonneaub, F. Mariettea, A. Davenela, and M. Kouba. Quantification of muscle, subcutaneous fat and intermuscular fat in pig carcasses and cuts by magnetic resonance imaging. *Meat Science*, 72:146–154, 2006.

[90] S. G. Nash. A survey of truncated-newton methods. *J. Comput. Appl. Math.*, 124(1-2): 45–59, 2000. ISSN 0377-0427.

[91] EA Navajas, CA Glasbey, KA McLean, AV Fisher, AJL Charteris, NR Lambe, L. Bünger, and G. Simm. In vivo measurements of muscle volume by automatic image analysis of spiral computed tomography scans. *Animal Science*, 82(04):545–553, 2007.

[92] P. M. Nissen, H. Busk, M. Oksama, M. Seynaeve, M. Gispert, P. Walstra, I. Hansson, and E. Olsen. The estimated accuracy of the EU reference dissection method for pig carcass classification. *Meat Science*, 73(1):22–28, 2006.

[93] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2nd edition, 2006.

[94] H. Ólafsdóttir. *Analysis of Craniofacial Images using Computational Atlases and Deformation Fields*. PhD thesis, Technical University of Denmark, apr 2008.

[95] H. Olafsdottir, T.A. Darvann, N.V. Hermann, E. Oubel, B.K. Ersboll, A.F. Frangi, P. Larsen, C.A. Perlyn, G.M. Morriss-Kay, and S. Kreiborg. Computational mouse atlases and their application to automatic assessment of craniofacial dysmorphology caused by the Crouzon mutation Fgfr2C342Y. *Journal of Anatomy*, 211(1):37, 2007.

[96] E. V. Olsen. *Encyclopedia of Meat Sciences*, volume 1. Academic Press, 2004.

[97] E. V. Olsen, M. Candek-Potokar, M. Oksama, S. Kien, D. Lisiak, and H. Busk. On-line measurements in pig carcass classification: Repeatability and variation caused by the operator and the copy of instrument. *Meat Science*, 75(1):29–38, 2007.

[98] V. Pereyra and G. Scherer. Efficient computer manipulation of tensor products with applications to multidimensional approximation. *Math. Comput*, 27(123):595–605, 1973.

[99] S. Pieper, B. Lorensen, W. Schroeder, and R. Kikinis. The na-mic kit: Itk, vtk, pipelines, grids and 3d slicer as an open platform for the medical image computing community. In *Biomedical Imaging: Nano to Macro, 2006. 3rd IEEE International Symposium on*, pages 698–701, 2006.

[100] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual-information-based registration of medical images: a survey. *IEEE Transactions on Medical Imaging*, 22 (8):986–1004, 2003.

[101] N. Roberts, L. M. Cruz-Orive, N. M. Reid, D. A. Brodie, M. Bourne, and R. H. Edwards. Unbiased estimation of human body composition by the cavalieri method using magnetic resonance imaging. *Journal of Microscopy*, 171:239–253, 1993.

[102] T. Rohlfing and C.R. Maurer. Multi-classifier framework for atlas-based image segmentation. *Pattern Recognition Letters*, 26(13):2070–2079, 2005.

[103] T. Rohlfing, CR Maurer Jr, DA Bluemke, and MA Jacobs. Volume-preserving nonrigid registration of MR breast images using free-form deformation with an incompressibility constraint. *Medical Imaging, IEEE Transactions on*, 22(6):730–741, 2003.

[104] T. Rohlfing, R. Brandt, R. Menzel, D. B. Russakoff, and C. R. Maurer Jr. *Handbook of Biomedical Image Analysis Volume III: Registration Models*, chapter Quo Vadis , Atlas-Based Segmentation?, pages 435–486. Springer, 2005.

[105] K. Rohr. *Landmark-Based Image Analysis: Using Geometric and Intensity Models*. Kluwer Academic Publishers, 2001.

[106] R. Romvári, A. Dobrowolski, I. Repa, P. Allen, E. Olsen, A. Szabo, and P. Horn. Development of a computed tomography calibration method for the determination of lean meat content in pig carcasses. *Acta Veterinaria Hungarica*, 54(1):1–10, 2006.

[107] A. Rosenfeld, editor. *Multiresolution Image Processing and Analysis*. Springer, Berlin, 1984. Leesberg, VA, July 1982.

[108] D. Rueckert, L.I. Sonoda, C. Hayes, D.L.G. Hill, M.O. Leach, and D.J. Hawkes. Nonrigid registration using free-form deformations: Application to breast mr images. *MedImg*, 18(8):712–721, August 1999.

[109] D. Rueckert, A.F. Frangi, and J.A. Schnabel. Automatic construction of 3-D statistical deformation models of the brain using nonrigid registration. *IEEE Trans. Medical Imaging*, 22(8):1014–1025, August 2003.

[110] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2003.

[111] J. A. Schnabel, D. Rueckert, M. Quist, J. M. Blackall, A. D. Castellano Smith, T. Hartkens, G. P. Penney, W. A. Hall, H. Liu, C. L. Truwit, F. A. Gerritsen, D. L. G. Hill, and D. J. Hawkes. A generic framework for non-rigid registration based on non-uniform multi-level free-form deformations. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI'01*, Lecture Notes in Computer Science, pages 573–581. Springer, 2001.

[112] H. Skjervold, K. Grønseth, O. Vangen, and A. Evensen. In vivo estimation of body composition by computerized tomography. *Zeitschrift für Tierzüchtung und Züchtungsbiologie*, 98:77–79, 1981.

[113] Danske Slagterier. Statistik 2007 - Svin (in Danish). Technical report, Danish Meat Association, 2008. URL www.danishmeat.dk.

[114] Danske Slagterier. Årsberetning 2006-07 (in Danish). Technical report, Danish Meat Association, 2007. URL www.danishmeat.dk.

[115] K. Stanford, S.D.M. Jones, and MA Price. Methods of predicting lamb carcass composition: A review. *Small Ruminant Research*, 29(3):241–254, 1998.

[116] M. B. Stegmann, B. K. Ersbøll, and R. Larsen. FAME - A Flexible Appearance Modelling Environment. *IEEE Transactions on Medical Imaging*, 22(10):1319–1331, 2003.

[117] C. Studholme. Simultaneous population based image alignment for template free spatial normalisation of brain anatomy. In *Biomedical Image Registration*, pages 81–90. Springer, 2003.

[118] C. Studholme, D. L. G. Hill, and D. J. Hawkes. An overlap invariant entropy measure of 3D medical image alignment. *Pattern Recognition*, 32(1):71–86, 1999.

[119] Cs. Szabo, L. Babinszky, M. W. A. Verstegen, O. Vangen, A. J. M. Jansman, and E. Kanis. The application of digital imaging techniques in the in vivo estimation of the body composition of pigs: a review. *Livestock Production Science*, 60(1):1–11, 1999.

[120] P. Thévenaz and M. Unser. Optimization of mutual information for multiresolution image registration. *IEEE Transactions on Image Processing*, 9(12):2083–2099, December 2000.

[121] P. Thévenaz, T. Blu, and M. Unser. Image interpolation and resampling. In I.N. Bankman, editor, *Handbook of Medical Imaging, Processing and Analysis*, chapter 25, pages 393–420. Academic Press, San Diego CA, USA, 2000.

[122] J.P. Thirion. Non-Rigid Matching Using Demons. In *IEEE Computer Society Conference On Computer Vision And Pattern Recognition*, pages 245–251. Institute Of Electrical Engineers Inc (IEEE), 1996.

[123] J.P. Thirion. Image matching as a diffusion process: an analogy with Maxwell's demons. *Medical Image Analysis*, 2(3):243–260, 1998.

[124] P.M. Thompson, M.S. Mega, K.L. Narr, E.R. Sowell, R.E. Blanton, and A.W. Toga. *Brain image analysis and atlas construction*, volume 2, chapter 17, pages 1061–1129. SPIE, 2000.

[125] A.N. Tikhonov and V.Y. Arsenin. Solutions of Ill-Posed Problems. . *VH Winston & Sons. Washington, DC*, 1977.

[126] T.Vercauteren, X. Pennec, A. Perchant, and N. Ayache. Non-parametric diffeomorphic image registration with the demons algorithm. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2007*. Springer, 2007.

[127] C. J. Twining, T. Cootes, S. Marsland, V. Petrovic, . Schestowitz, and C. J. Taylor. A unified information-theoretic approach to groupwise non-rigid registration and model building. In *Information Processing in Medical Imaging*, pages 1–14. Springer, 2005.

[128] M. Unser. Ten good reasons for using spline wavelets. In *Proceedings of the SPIE Conference on Mathematical Imaging: Wavelet Applications in Signal and Image Processing V*, volume 3169, pages 422–431, San Diego CA, USA, July 30-August 1, 1997.

[129] C.F. Van Loan. The ubiquitous kronecker product. *Journal of Computational and Applied Mathematics*, 123(1-2):85–100, 2000.

[130] M. Vester-Christensen, L. B. Christensen, M. Hviid, E. V. Olsen, and R. Larsen. Virtual jointing of pig carcasses using image registration. Technical report, Department of Informatics and Mathematical Modelling. Technical University of Denmark, 2008.

[131] M. Vester-Christensen, S. G. Erbou, M. F. Hansen, E. V. Olsen, L. B. Christensen, M. Hviid, B. K. Ersbøll, and R. Larsen. Virtual dissection of pig carcasses. *Meat Science*, 2008. Under review.

[132] C. Vestergaard, S. G. Erbou, T. Thauland, J. Adler-Nissen, and P. Berg. Salt distribution in dry-cured ham measured by computed tomography and image analysis. *Meat Science*, 69:9–15, 2005.

[133] P.A. Viola. *Alignment by Maximization of Mutual Information*. PhD thesis, Massachusetts Institute of Technology, 1995.

[134] P. Walstra and G. S. M. Merkus. Procedure for assesment of the lean meat percentage as a consequence of the new EU reference dissection method in pig carcass classification. ID-DLO 96.014, 1996.

[135] R.P. Woods, S.T. Grafton, J.D.G. Watson, N.L. Sicotte, and J.C. Mazziotta. Automated Image Registration: II. Intersubject Validation of Linear and Nonlinear Models. *Journal of Computer Assisted Tomography*, 22(1):153, 1998.

[136] Z. Xie and G. E. Farin. Image registration using hierarchical b-splines. *IEEE Transactions on Visualization and Computer Graphics*, 10(1):85–94, 2004. ISSN 1077-2626. doi: http://dx.doi.org.globalproxy.cvt.dk/10.1109/TVCG.2004.1260760.

[137] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000, 2003.